

LNCS 3115

Peter Enser  
Yiannis Kompatsiaris  
Noel E. O'Connor  
Alan F. Smeaton  
Arnold W.M. Smeulders (Eds.)

# Image and Video Retrieval

Third International Conference, CIVR 2004  
Dublin, Ireland, July 2004  
Proceedings

 Springer

*Commenced Publication in 1973*

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

## Editorial Board

Takeo Kanade

*Carnegie Mellon University, Pittsburgh, PA, USA*

Josef Kittler

*University of Surrey, Guildford, UK*

Jon M. Kleinberg

*Cornell University, Ithaca, NY, USA*

Friedemann Mattern

*ETH Zurich, Switzerland*

John C. Mitchell

*Stanford University, CA, USA*

Moni Naor

*Weizmann Institute of Science, Rehovot, Israel*

Oscar Nierstrasz

*University of Bern, Switzerland*

C. Pandu Rangan

*Indian Institute of Technology, Madras, India*

Bernhard Steffen

*University of Dortmund, Germany*

Madhu Sudan

*Massachusetts Institute of Technology, MA, USA*

Demetri Terzopoulos

*New York University, NY, USA*

Doug Tygar

*University of California, Berkeley, CA, USA*

Moshe Y. Vardi

*Rice University, Houston, TX, USA*

Gerhard Weikum

*Max-Planck Institute of Computer Science, Saarbruecken, Germany*

Peter Enser Yiannis Kompatsiaris  
Noel E. O'Connor Alan F. Smeaton  
Arnold W.M. Smeulders (Eds.)

# Image and Video Retrieval

Third International Conference, CIVR 2004  
Dublin, Ireland, July 21-23, 2004  
Proceedings



Springer

## Volume Editors

Peter Enser  
University of Brighton  
School of Computing, Mathematical and Information Sciences  
Watts Building, Moulsecoomb, Brighton BN2 4GJ, UK  
E-mail: pgbe@bton.ac.uk

Yiannis Kompatsiaris  
Centre for Research and Technology - Hellas  
Informatics and Telematics Institute (CERTH/ITI)  
1st km. Thermi-Panorama Road, P.O. Box 361, 57001 Thermi-Thessaloniki, Greece  
E-mail: ikom@iti.gr

Noel E. O'Connor  
Alan F. Smeaton  
Dublin City University  
Centre for Digital Video Processing  
Collins Ave, Dublin 9, Ireland  
E-mail: oconnorn@eeng.dcu.ie  
ASmeaton@computing.dcu.ie

Arnold W.M. Smeulders  
University of Amsterdam  
ISIS group, Informatics Institute  
Kruislaan 403, 1098SJ Amsterdam, The Netherlands  
E-mail: smeulders@science.uva.nl

Library of Congress Control Number: 2004108946

CR Subject Classification (1998): H.3, H.2, H.4, H.5.1, H.5.4-5, I.4

ISSN 0302-9743

ISBN 3-540-22539-0 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable to prosecution under the German Copyright Law.

Springer-Verlag is a part of Springer Science+Business Media  
springeronline.com

© Springer-Verlag Berlin Heidelberg 2004  
Printed in Germany

Typesetting: Camera-ready by author, data conversion by PTP-Berlin, Protago-TeX-Production GmbH  
Printed on acid-free paper      SPIN: 11300175      06/3142      5 4 3 2 1 0



## Preface

We greeted the attendees of CIVR 2004 with the following address: “Táimid an-bhrodúil fáilte a chur romhaibh chuig Ollscoil Chathair Bhaile Átha Cliath agus chuig an triú Comhdháil Idirnáisiúnta ar Aisghabháil Íomhánna agus Físeán. Tá súil againn go mbeidh am iontach agaibh anseo in Éirinn agus go mbeidh bhur gcúairt taitneamhnach agus sásúil. Táimid an-bhrodúil go háirithe fáilte a chur roimh na daoine ón oiread sin tíortha difriúla agus na daoine a tháinig as i bhfad i gcéin. Tá an oiread sin páipéar curtha isteach chuig an chomhdháil seo go bhfuil caighdeán na bpáipéar agus na bpóstaer an-ard ar fad agus táimid ag súil go mór le hócaid iontach.”

We were delighted to host the 3<sup>rd</sup> International Conference on Image and Video Retrieval in Dublin City University. We hope that all attendees had a wonderful stay in Ireland and that their visits were enjoyable and rewarding.

There were 125 papers in total submitted to the CIVR 2004 conference and each was reviewed by at least three independent reviewers. We are grateful to the 64 members of the technical programme committee and the 29 other reviewers who completed these reviews and allowed us to put together a very strong technical programme. The programme included 4 invited keynote presentations from Nick Belkin, Shih-Fu Chang, Andrew Fitzgibbon and Joe Marks and we are very grateful to them for their incisive and thoughtful presentations. The programme also contained 29 paper presentations and 44 poster presentations as well as two other guest presentations. The programme committee chairs did an excellent job in putting together the schedule for this conference, and the local arrangements, finance and publicity chairs also put a lot of work into making this conference happen. Special thanks should go to Cathal Gurrin and Hyowon Lee for the tremendous amount of hard work they put into supporting CIVR 2004.

The CIVR 2004 conference was held in cooperation with the ACM SIGIR, the Irish Pattern Recognition and Classification Society, the EU FP5 SCHEMA Network of Excellence, and the BCS IRSG, and we are grateful to these organizations for promoting the event. We are also indebted to Science Foundation Ireland whose financial support allowed many students and other delegates to attend.

June 2004

Alan F. Smeaton  
Noel E. O'Connor

# International Conference on Image and Video Retrieval 2004 Organization

## Organizing Committee

General Chair	Alan F. Smeaton (Dublin City University)
Program Co-chairs	Noel E. O'Connor (Dublin City University)
	Arnold Smeulders (University of Amsterdam)
Practitioner Co-chairs	Peter Enser (University of Brighton)
	Yiannis Kompatsiaris (Informatics and Telematics Institute, Greece)
Finance	Noel Murphy (Dublin City University)
Local Arrangements	Gareth Jones (Dublin City University)
	Seán Marlow (Dublin City University)
Publicity	Cathal Gurrin (Dublin City University)
	Hyowon Lee (Dublin City University)

## Programme Committee

Kiyo Aizawa, University of Tokyo, Japan  
Erwin M. Bakker, Leiden University, The Netherlands  
Alberto Del Bimbo, University of Florence, Italy  
Nevenka Dimitrova, Philips Research, USA  
David M. Blei, University of California, Berkeley, USA  
Patrick Bouthemy, IRISA/INRIA, France  
Josep R. Casas, Technical University of Catalonia (UPC), Spain  
Shih-Fu Chang, Columbia University, USA  
Tsuhan Chen, Carnegie Mellon University, USA  
John Eakins, University of Northumbria, UK  
Graham Finlayson, University of East Anglia, UK  
Andrew Fitzgibbon, University of Oxford, UK  
David Forsyth, University of California, Berkeley, USA  
Theo Gevers, University of Amsterdam, The Netherlands  
Abby Goodrum, Syracuse University, USA  
Patrick Gros, IRISA/CNRS, France  
Alan Hanjalic, Delft University of Technology, The Netherlands  
Richard Harvey, University of East Anglia, UK  
Paola Hobson, Motorola Research Labs, UK  
Thomas Huang, University of Illinois at Urbana-Champaign, USA  
Ichiro Ide, National Institute of Informatics, Japan  
Horace Ip, City University of Hong Kong  
Ebroul Izquierdo, Queen Mary University of London, UK

## VIII Organization

Alejandro Jaimes, Fuji Xerox Nakai Research Center, Japan  
Ramesh Jain, Georgia Tech, USA  
Joemon J. Jose, Glasgow University, UK  
Avi Kak, Purdue University, USA  
Josef Kittler, University of Surrey, UK  
Anil Kokaram, Trinity College Dublin, Ireland  
Wessel Kraaij, TNO, The Netherlands  
Clement Leung, Victoria University, Melbourne, Australia  
Michael Lew, Leiden University, The Netherlands  
Paul Lewis, University of Southampton, UK  
R. Manmatha, University of Massachusetts, USA  
Stéphane Marchand-Maillet, University of Geneva, Switzerland  
Jiri (George) Matas, CVUT Prague, Czech Republic  
Bernard Merialdo, Eurecom, France  
Philippe Mulhem, CLIPS-IMAG, France  
Milind Naphade, IBM T.J. Watson Research Center, USA  
Jan Nesvadba, Philips Research, Eindhoven, The Netherlands  
Eric Pauwels, CWI, The Netherlands  
Dragutin Petkovic, San Francisco State University, USA  
Matthias Rauterberg, Technical University Eindhoven, The Netherlands  
Stefan Rüger, Imperial College London, UK  
Yong Rui, Microsoft Research, USA  
Andrew Salway, University of Surrey, UK  
Stan Sclaroff, Boston University, USA  
Nicu Sebe, University of Amsterdam, The Netherlands  
Ibrahim Sezan, Sharp Labs of America, USA  
Thomas Sikora, Technical University of Berlin, Germany  
John R. Smith, IBM Research, USA  
Michael Strintzis, Informatics and Telematics Institute, Greece  
Sanghoon Sull, Korea University, Korea  
Tieniu Tan, Chinese Academy of Sciences, China  
Qi Tian, University of Texas at San Antonio, USA  
Belle Tseng, IBM T.J. Watson Research Center, USA  
Arjen P. de Vries, CWI, The Netherlands  
Thijs Westerveld, CWI, The Netherlands  
Marcel Worring, University of Amsterdam, The Netherlands  
Guangyou Xu, Tsinghua University, China  
Li-Qun Xu, BT Exact, UK  
HongJiang Zhang, Microsoft Research China  
Xiang (Sean) Zhou, Siemens Corporate Research, USA  
Andrew Zisserman, University of Oxford, UK

## Additional Reviewers

Tomasz Adamek, Dublin City University, Ireland  
Lalitha Agnihotri, Philips Research, USA  
Stephen Blott, Dublin City University, Ireland  
Paul Browne, Dublin City University, Ireland  
Orla Duffner, Dublin City University, Ireland  
Jun Fan, Philips Research, USA  
S.L. Feng, University of Massachusetts, USA  
Georgina Gaughan, Dublin City University, Ireland  
Jiwoon Jeon, University of Massachusetts, USA  
Dongge Li, Motorola Labs, USA  
Jovanka Malobabic, Dublin City University, Ireland  
Kieran McDonald, Dublin City University, Ireland  
Hiroshi Mo, National Institute of Informatics, Japan  
Ciaran O’Conaire, Dublin City University, Ireland  
Neil O’Hare, Dublin City University, Ireland  
Kadir A. Peker, Mitsubishi Electric Research Labs, USA  
Regunathan Radhakrishnan, Mitsubishi Electric Research Labs, USA  
Toni Rath, University of Massachusetts, USA  
Gus Reid, Motorola Research Labs, UK  
Robert Sadleir, Dublin City University, Ireland  
David Sadlier, Dublin City University, Ireland  
Sorin Sav, Dublin City University, Ireland  
Alistair Sutherland, Dublin City University, Ireland  
Jonathan Teh, Motorola Research Labs, UK  
Simon Waddington, Motorola Research Labs, UK  
Yi Wu, University of California at Santa Barbara, USA  
Qing Xue, University of Texas at San Antonio, USA  
Jiamin Ye, Dublin City University, Ireland  
Jerry Jie Yu, University of Texas at San Antonio, USA

Sponsors

CIVR 2004 was organized by the Centre for Digital Video Processing at Dublin City University. The event was co-sponsored by:



and held in co-operation with:



# Table of Contents

## Keynote Speaker Abstracts

Pattern Mining in Large-Scale Image and Video Sources . . . . .	1
<i>Shih-Fu Chang</i>	
Computer Vision in the Movies: From the Lab to the Big Screen . . . . .	2
<i>Andrew Fitzgibbon</i>	
Image and Video Retrieval Using New Capture and Display Devices . . . . .	3
<i>Joe Marks</i>	
Models of Interaction with Video Information . . . . .	5
<i>Nicholas J. Belkin</i>	

## Image Annotation and User Searching

User Strategies in Video Retrieval: A Case Study . . . . .	6
<i>L. Hollink, G.P. Nguyen, D.C. Koelma, A.T. Schreiber, M. Worring</i>	
Video Content Foraging . . . . .	15
<i>Ynze van Houten, Jan Gerrit Schuurman, Pløn Verhagen</i>	
Using Maximum Entropy for Automatic Image Annotation . . . . .	24
<i>Jiwoon Jeon, R. Manmatha</i>	
Everything Gets Better All the Time, Apart from the Amount of Data . . . . .	33
<i>Hieu T. Nguyen, Arnold Smeulders</i>	

## Image and Video Retrieval Algorithms (I)

An Inference Network Approach to Image Retrieval . . . . .	42
<i>Donald Metzler, R. Manmatha</i>	
Small Sample Size Performance of Evolutionary Algorithms for Adaptive Image Retrieval . . . . .	51
<i>Zoran Stejić, Yasufumi Takama, Kaoru Hirota</i>	
Co-retrieval: A Boosted Reranking Approach for Video Retrieval . . . . .	60
<i>Rong Yan, Alexander G. Hauptmann</i>	

# Poster Session (I)

HMM Model Selection Issues for Soccer Video .....	70
<i>Mark Baillie, Joemon M. Jose, Cornelis J. van Rijsbergen</i>	
Tennis Video Analysis Based on Transformed Motion Vectors.....	79
<i>Peng Wang, Rui Cai, Shi-Qiang Yang</i>	
Semantic Event Detection in Sports Through Motion Understanding .....	88
<i>N. Rea, R. Dahyot, A. Kokaram</i>	
Structuring Soccer Video Based on Audio Classification and Segmentation Using Hidden Markov Model .....	98
<i>Jianyun Chen, Yunhao Li, Songyang Lao, Lingda Wu, Liang Bai</i>	
EDU: A Model of Video Summarization .....	106
<i>Yu-Xiang Xie, Xi-Dao Luan, Song-Yang Lao, Ling-Da Wu, Peng Xiao, Jun Wen</i>	
A News Video Mining Method Based on Statistical Analysis and Visualization .....	115
<i>Yu-Xiang Xie, Xi-Dao Luan, Song-Yang Lao, Ling-Da Wu, Peng Xiao, Zhi-Guang Han</i>	
Topic Threading for Structuring a Large-Scale News Video Archive.....	123
<i>Ichiro Ide, Hiroshi Mo, Norio Katayama, Shin'ichi Satoh</i>	
What's News, What's Not? Associating News Videos with Words .....	132
<i>Pinar Duygulu, Alexander Hauptmann</i>	
Visual Clustering of Trademarks Using a Component-Based Matching Framework .....	141
<i>Mustaq Hussain, John P. Eakins</i>	
Assessing Scene Structuring in Consumer Videos .....	150
<i>Daniel Gatica-Perez, Napat Triroj, Jean-Marc Odobez, Alexander Loui, Ming-Ting Sun</i>	
A Visual Model Approach for Parsing Colonoscopy Videos .....	160
<i>Yu Cao, Wallapak Tavanapong, Dalei Li, JungHwan Oh, Piet C. de Groen, Johnny Wong</i>	
Video Summarization and Retrieval System Using Face Recognition and MPEG-7 Descriptors .....	170
<i>Jae-Ho Lee, Whoi-Yul Kim</i>	
Automatic Generation of Personalized Digest Based on Context Flow and Distinctive Events .....	179
<i>Hisashi Miyamori</i>	

Content-Based Image Retrieval and Characterization on Specific Web Collections .....	189
<i>R. Baeza-Yates, J. Ruiz-del-Solar, R. Verschae, C. Castillo, C. Hurtado</i>	
Exploiting Problem Domain Knowledge for Accurate Building Image Classification .....	199
<i>Andres Dorado, Ebroul Izquierdo</i>	
Natural Scene Retrieval Based on a Semantic Modeling Step .....	207
<i>Julia Vogel, Bernt Schiele</i>	
Unsupervised Text Segmentation Using Color and Wavelet Features .....	216
<i>Julinda Gllavata, Ralph Ewerth, Teuta Stefi, Bernd Freisleben</i>	
Universal and Personalized Access to Content via J2ME Terminals in the DYMAS System .....	225
<i>Ana García, José M. Martínez, Luis Herranz</i>	
Task-Based User Evaluation of Content-Based Image Database Browsing Systems .....	234
<i>Timo Ojala, Markus Koskela, Esa Matinmikko, Mika Rautiainen, Jorma Laaksonen, Erkki Oja</i>	
The CLEF Cross Language Image Retrieval Track (ImageCLEF) 2004 ...	243
<i>Paul Clough, Mark Sanderson, Henning Müller</i>	
An Empirical Investigation of the Scalability of a Multiple Viewpoint CBIR System .....	252
<i>James C. French, Xiangyu Jin, W.N. Martin</i>	
Real-Time Video Indexing System for Live Digital Broadcast TV Programs .....	261
<i>Ja-Cheon Yoon, Hyeokman Kim, Seong Soo Chun, Jung-Rim Kim, Sanghoon Sull</i>	
<b>Person and Event Identification for Retrieval</b>	
Finding Person X: Correlating Names with Visual Appearances .....	270
<i>Jun Yang, Ming-yu Chen, Alex Hauptmann</i>	
A Framework for Semantic Classification of Scenes Using Finite State Machines .....	279
<i>Yun Zhai, Zeeshan Rasheed, Mubarak Shah</i>	
Automated Person Identification in Video .....	289
<i>Mark Everingham, Andrew Zisserman</i>	



## Content-Based Image and Video Retrieval (I)

Content Based Image Synthesis .....	299
<i>Nicholas Diakopoulos, Irfan Essa, Ramesh Jain</i>	
Interactive Content-Based Retrieval Using Pre-computed Object-Object Similarities .....	308
<i>Liudmila Boldareva, Djoerd Hiemstra</i>	

## Content-Based Image and Video Retrieval (II)

Salient Regions for Query by Image Content.....	317
<i>Jonathon S. Hare, Paul H. Lewis</i>	
Evaluation of Texture Features for Content-Based Image Retrieval .....	326
<i>Peter Howarth, Stefan Rüger</i>	
An Effective Approach Towards Content-Based Image Retrieval.....	335
<i>Rokia Missaoui, M. Sarifuddin, Jean Vaillancourt</i>	

## Image and Video Retrieval Algorithms (II)

Multimedia Retrieval Using Multiple Examples .....	344
<i>Thijs Westerveld, Arjen P. de Vries</i>	
A Discussion of Nonlinear Variants of Biased Discriminants for Interactive Image Retrieval.....	353
<i>Xiang Sean Zhou, Ashutosh Garg, Thomas S. Huang</i>	

## Poster Session (II)

Salient Objects: Semantic Building Blocks for Image Concept Interpretation .....	365
<i>Jianping Fan, Yuli Gao, Hangzai Luo, Guangyou Xu</i>	
Multimodal Salient Objects: General Building Blocks of Semantic Video Concepts .....	374
<i>Hangzai Luo, Jianping Fan, Yuli Gao, Guangyou Xu</i>	
Video Content Representation as Salient Regions of Activity .....	384
<i>Nicolas Moënné-Loccoz, Eric Bruno, Stéphane Marchand-Maillet</i>	
Image Classification into Object / Non-object Classes .....	393
<i>Sungyoung Kim, Sojung Park, Minhwan Kim</i>	
Video Segmentation Using Hidden Markov Model with Multimodal Features.....	401
<i>Tae Meon Bae, Sung Ho Jin, Yong Man Ro</i>	

Feature Based Cut Detection with Automatic Threshold Selection . . . . .	410
<i>Anthony Whitehead, Prosenjit Bose, Robert Laganieri</i>	
A Geometrical Key-Frame Selection Method	
Exploiting Dominant Motion Estimation in Video . . . . .	419
<i>Brigitte Fauvet, Patrick Bouthemy, Patrick Gros, Fabien Spindler</i>	
Extraction of Salient Features for Image Retrieval	
Using Multi-scale Image Relevance Function . . . . .	428
<i>Roman M. Palenichka, Rokia Missaoui, Marek B. Zaremba</i>	
Relevance Feedback for Keyword and Visual	
Feature-Based Image Retrieval . . . . .	438
<i>Feng Jing, Mingjing Li, Hong-Jiang Zhang, Bo Zhang</i>	
Relevance Feedback Reinforced with Semantics Accumulation . . . . .	448
<i>Sangwook Oh, Min Gyo Chung, Sanghoon Sull</i>	
Faster Exact Histogram Intersection on Large Data Collections	
Using Inverted VA-Files . . . . .	455
<i>Wolfgang Müller, Andreas Henrich</i>	
A Graph Edit Distance Based on Node Merging . . . . .	464
<i>S. Berretti, A. Del Bimbo, P. Pala</i>	
STRICT: An Image Retrieval Platform for Queries	
Based on Regional Content . . . . .	473
<i>Jean-Francois Omhover, Marcin Detyniecki</i>	
Improved Video Content Indexing	
by Multiple Latent Semantic Analysis . . . . .	483
<i>Fabrice Souvannavong, Bernard Merialdo, Benoît Huet</i>	
Three Interfaces for Content-Based Access to Image Collections . . . . .	491
<i>Daniel Heesch, Stefan Rüger</i>	
Retrieving ClipArt Images by Content . . . . .	500
<i>Manuel J. Fonseca, B. Barroso, P. Ribeiro, Joaquim A. Jorge</i>	
Use of Image Subset Features in Image Retrieval	
with Self-Organizing Maps . . . . .	508
<i>Markus Koskela, Jorma Laaksonen, Erkki Oja</i>	
An Indexing Model of Remote Sensing Images . . . . .	517
<i>Paola Carrara, Gabriella Pasi, Monica Pepe, Anna Rampini</i>	
Ambient Intelligence Through Image Retrieval . . . . .	526
<i>Jean-Marc Seigneur, Daniel Solis, Fergal Shevlin</i>	

A Knowledge Management System for Intelligent Retrieval of Geo-spatial Imagery . . . . .	535
<i>Eoin McLoughlin, Dymrna O'Sullivan, Michela Bertolotto, David Wilson</i>	

An Adaptive Image Content Representation and Segmentation Approach to Automatic Image Annotation . . . . .	545
<i>Rui Shi, Huamin Feng, Tat-Seng Chua, Chin-Hui Lee</i>	

Knowledge Assisted Analysis and Categorization for Semantic Video Retrieval . . . . .	555
<i>Manolis Wallace, Thanos Athanasiadis, Yannis Avrithis</i>	

## Content-Based Image and Video Retrieval (III)

Using Structure for Video Object Retrieval . . . . .	564
<i>Lukas Hohl, Fabrice Souvannavong, Bernard Meriardo, Benoît Huet</i>	

Object Segmentation and Ontologies for MPEG-2 Video Indexing and Retrieval . . . . .	573
<i>Vasileios Mezaris, Michael G. Strintzis</i>	

Interoperability Support for Ontology-Based Video Retrieval Applications . . . . .	582
<i>Chrisa Tsinaraki, Panagiotis Polydoros, Stavros Christodoulakis</i>	

## EU Project Session (I)

A Test-Bed for Region-Based Image Retrieval Using Multiple Segmentation Algorithms and the MPEG-7 eXperimentation Model: The Schema Reference System . . . . .	592
<i>Vasileios Mezaris, Haralambos Doulaverakis, Raul Medina Beltran de Otalora, Stephan Herrmann, Ioannis Kompatsiaris, Michael G. Strintzis</i>	

ICBR – Multimedia Management System for Intelligent Content Based Retrieval . . . . .	601
<i>Janko Čalić, Neill Campbell, Majid Mirmehdi, Barry T. Thomas, Ron Laborde, Sarah Porter, Nishan Canagarajah</i>	

Contribution of NLP to the Content Indexing of Multimedia Documents . . . . .	610
<i>Thierry Declerck, Jan Kuper, Horacio Saggion, Anna Samiotou, Peter Wittenburg, Jesus Contreras</i>	

The CIMWOS Multimedia Indexing System . . . . .	619
<i>Harris Papageorgiou, Athanassios Protopapas</i>	

## User Perspectives

Image Retrieval Interfaces: A User Perspective .....	628
<i>John P. Eakins, Pam Briggs, Bryan Burford</i>	

## EU Project Session (II)

SCULPTEUR: Multimedia Retrieval for Museums .....	638
<i>Simon Goodall, Paul H. Lewis, Kirk Martinez, Patrick A.S. Sinclair, Fabrizio Giorgini, Matthew J. Addis, Mike J. Boniface, Christian Lahanier, James Stevenson</i>	

Disclosure of Non-scripted Video Content: InDiCo and M4/AMI .....	647
<i>Franciska de Jong</i>	

A User-Centred System for End-to-End Secure Multimedia Content Delivery: From Content Annotation to Consumer Consumption .....	656
<i>Li-Qun Xu, Paulo Villegas, Mónica Díez, Ebroul Izquierdo, Stephan Herrmann, Vincent Bottreau, Ivan Damnjanovic, Damien Papworth</i>	

Adding Semantics to Audiovisual Content: The FAETHON Project .....	665
<i>Thanos Athanasiadis, Yannis Avrithis</i>	

Towards a Large Scale Concept Ontology for Broadcast Video .....	674
<i>Alexander G. Hauptmann</i>	

Author Index .....	677
--------------------	-----

# Pattern Mining in Large-Scale Image and Video Sources

Shih-Fu Chang

Digital Video and Multimedia Lab, Department of Electrical Engineering,  
Columbia University, New York, NY 10027, USA  
sfchang@ee.columbia.edu  
<http://www.ee.columbia.edu/dvmm>  
<http://www.ee.columbia.edu/~sfchang>

**Abstract.** Detection and recognition of semantic events has been a major research challenge for multimedia indexing. An emerging direction in this field has been *unsupervised discovery (mining) of patterns* in spatial-temporal multimedia data. Patterns are recurrent, predictable occurrences of one or more entities that satisfy statistical, associative, or relational conditions. Patterns at the feature level may signify the occurrence of primitive events (e.g., recurrent passing of pedestrians). At the higher level, patterns may represent cross-event relations; e.g., recurrent news stories across multiple broadcast channels or repetitive play-break alternations in sports. Patterns in an annotated image collection may indicate collocations of related semantic concepts and perceptual clusters.

Mining of patterns of different types at different levels offers rich benefits, including automatic discovery of salient events or topics in a new domain, automatic generation of alerts indicating unusual situations, and summarization of concepts structures in a massive collection of content.

Many challenging issues emerge. What are the adequate representations and statistical models for patterns that may exist at different levels and different time scales? How do we effectively detect and fuse patterns supported by different media modalities, as well as how to handle patterns that may have relatively sparse occurring frequencies? How do we evaluate the quality of mining results given its unsupervised nature?

In this talk, I will present results of our recent efforts in mining patterns in structured video sequences (such as sports and multi-channel broadcast news) and large collection of stock photos. Specifically, we will discuss the potential of statistical models like Hierarchical HMM for temporal structure mining, probabilistic latent semantic analysis for discovering hidden concepts, a hierarchical mixture model for fusing multi-modal patterns, and the combined exploration of electronic knowledge (such as WordNet) and statistical clustering for image knowledge mining.

Evaluations against real-world videos such as broadcast sports, multi-channel news, and stock photos will be presented. Future directions and open issues will be discussed.

# Computer Vision in the Movies: From the Lab to the Big Screen

Andrew Fitzgibbon

Visual Geometry Group, University of Oxford, U.K.  
awf@robots.ox.ac.uk  
<http://www.robots.ox.ac.uk/~awf>

**Abstract.** I will talk about recent and current work at Oxford on the automatic reconstruction of 3D information from 2D image sequences, and the applications of this technology to robotic navigation, augmented reality and the movie industry. The results of our work have been used on such movies as the „Lord of the Rings“ and „Harry Potter“ series, and in 2002 we received an Emmy award for our contributions to television.

The „take-home“ of this talk is in the story of how to move from leading-edge research ideas to reliable commercial-quality code which must perform under the rigid time constraints of the movie industry.

After an introduction to the basic tools of 3D reconstruction, I will demonstrate how the development of robust procedures for statistical estimation of geometric constraints from image sequences has led to the production of a highly reliable system based on leading-edge machine vision research. I will talk about the special character of film as an industry sector, and where that made our work easier or harder.

I shall conclude by talking about ongoing work in our lab, including vision and graphics as well as some visual information retrieval.

# Image and Video Retrieval Using New Capture and Display Devices

Joe Marks

Mitsubishi Electric Research Laboratories (MERL), Cambridge, Massachusetts, USA  
<http://www.merl.com>

**Abstract.** Given a standard camera and a standard display screen, image- and video-retrieval problems are well understood, if not yet solved. But what happens if the capture device is more capable? Or the display device? Some of the hard problems might well be made easier; and some of the impossible problems might become feasible.

In this talk I will survey several novel input and output devices being developed at MERL that have the ability to change the nature of image and video retrieval, especially for industrial applications. These projects include:

- *The Nonphotorealistic Camera:* By illuminating a scene with multiple flashes, discontinuity edges can be distinguished from texture edges. The identification of discontinuity edges allows for stylistic image renderings that suppress detail and enhance clarity.
- *The Fusion Camera:* Images are captured using two cameras. A regular video-camera captures visible light; another videocamera captures far-infrared radiation. The two image streams can be combined in novel ways to provide enhanced imagery.
- *Instrumenting Environments for Better Video Indexing:* Doors and furniture in work environments are instrumented with cheap ultrasonic transducers. The ultrasonic signals of these transducers are captured by microphone, down-shifted in frequency, and recorded on the normal audio track alongside the captured video. The recorded audio signals are used to help index or search through the video.
- *Visualizing Audio on Video:* A typical security guard can monitor up to 64 separate video streams simultaneously. However, it is impossible to monitor more than one audio stream at a time. Using a microphone array and sound-classification software, audio events can be identified and located. Visual representations of the audio events can then be overlaid on the normal video stream, thus giving the viewer some indication of the sounds in a video-monitored environment.
- *A Cheap Location-Aware Camera:* A videocamera is instrumented with a simple laser-projection system. Computer-vision techniques are used to determine the relative motion of the camera from the projected laser light. By relating the motion to a known fixed point, absolute camera location can be ob-

tained. The captured video can thus be indexed by camera location in a cost-effective manner.

- *An End-to-End 3D Capture-and-Display Video System*: Using a camera array, a projector array, and a lenticular screen, video can be captured and displayed in 3D without requiring viewers to wear special viewing glasses.
- *Image Browsing on a Multiuser Tabletop Display*: Many image-retrieval and image-analysis tasks may be performed better by teams of people than by individuals working alone. Novel hardware and software allow multiple users to view and manipulate imagery together on a tabletop display.

I will also speculate on how some future changes to capture and display devices may further impact image and video retrieval in the not-too-distant future.

*The projects above represent the research efforts of many members of the MERL staff. For appropriate credits, please visit the MERL web site, [www.merl.com](http://www.merl.com).*

#### Speaker bio:

Joe Marks grew up in Dublin, Ireland, before emigrating to the U.S. in 1979. He holds three degrees from Harvard University. His areas of interest include computer graphics, human-computer interaction, and artificial intelligence. He has worked previously at Bolt Beranek and Newman and at Digital's Cambridge Research Laboratory. He is currently the Director of MERL Research. He is also the recent past chair of ACM SIGART and the papers chair for SIGGRAPH 2004.



# Models of Interaction with Video Information

Nicholas J. Belkin

School of Communication, Information and Library Studies,  
Rutgers University, New Brunswick, NJ, USA  
[nick@belkin.rutgers.edu](mailto:nick@belkin.rutgers.edu)  
<http://www.scils.rutgers.edu/~belkin/belkin.html>

**Abstract.** A crucial issue for research in video information retrieval (VIR) is the relationship between the tasks which VIR is supposed to support, and the techniques of representation, matching, display and navigation within the VIR system which are most appropriate for responding to those tasks. Within a general model of information retrieval as support for user interaction with information objects, this paper discusses how different tasks might imply the use of different techniques, and in particular, different modes of interaction, for „optimal“ VIR within the different task environments. This analysis suggests that there will be no universally applicable VIR techniques, and that really effective VIR systems will necessarily be tailored to specific task environments. This in turn suggests that an important research agenda in VIR will be detailed task analyses, with concomitant specification of functionalities required to support people in accomplishment of their tasks.

# User Strategies in Video Retrieval: A Case Study

L. Hollink<sup>1</sup>, G.P. Nguyen<sup>2</sup>, D.C. Koelma<sup>2</sup>, A.T. Schreiber<sup>1</sup>, and M. Worring<sup>2</sup>

<sup>1</sup> Business Informatics, Free University Amsterdam. {hollink,schreiber}@cs.vu.nl

<sup>2</sup> ISIS, University of Amsterdam. {giangnp,koelma,worring}@science.uva.nl

**Abstract.** In this paper we present the results of a user study that was conducted in combination with a submission to TRECVID 2003. Search behavior of students querying an interactive video-retrieval system was analyzed. 242 Searches by 39 students on 24 topics were assessed. Questionnaire data, logged user actions on the system, and a quality measure of each search provided by TRECVID were studied. Analysis of the results at various stages in the retrieval process suggests that retrieval based on transcriptions of the speech in video data adds more to the average precision of the result than content-based retrieval. The latter is particularly useful in providing the user with an overview of the dataset and thus an indication of the success of a search.

## 1 Introduction

In this paper we present the results of a study in which search behavior of students querying an interactive video-retrieval system was analyzed. Recently, many techniques have been developed to automatically index and retrieve multimedia. The Video Retrieval Track at TREC (TRECVID) provides test collections and software to evaluate these techniques. Video data and statements of information need (topics) are provided in order to evaluate video-retrieval systems performing various tasks. In this way, the quality of the systems is measured. However, these measures give no indication of user performance. User variables like prior search experience, search strategies, and knowledge about the topic can be expected to influence the search results. Due to the recent nature of automatic retrieval systems, not many data are available about user experiences. We argue that knowledge about user behavior is one way to improve performance of retrieval systems. Interactive search in particular can benefit from this knowledge, since the user plays such a central role in the process.

We study information seeking behavior of users querying an interactive video-retrieval system. The study was conducted in combination with a submission to TRECVID 2003 [1]. Data were recorded about user characteristics, user estimations of the quality of their search results, familiarity of users with the topics, and actions performed while searching. The aim of the study was to investigate the influence of the recorded user variables on the average precision of the search results. In addition, a categorization was made of the 24 topics that were provided by TRECVID. The categories show differences in user behavior and average precision of the search results.

## 2 Research Questions

To gain knowledge about how user-related factors affect search in a state-of-the-art video-retrieval system, we record actions that users take when using such a system. In particular, we are interested in which actions lead to the best results. To achieve an optimal search result, it is important that a user knows when to stop searching. In this study we therefore measure how well users estimate the precision and recall of their search.

It is possible that different topics or categories of topics lead to different user strategies and differences in the quality of the results. We compare the search behavior and search results of categories of topics. In sum, the main questions in the study are:

1. What search actions are performed by users and which actions lead to the best search results?
2. Are users able to estimate the success of their search?
3. What is the influence of topic type on user actions and search results?

## 3 The ISIS Video Retrieval System

The video-retrieval system on which the study was performed was built by the Intelligent Sensory Information Systems (ISIS) group at the University of Amsterdam for the interactive video task at TRECVID. For a detailed description of the system we refer to [1].

The search process consists of four steps: indexing, filtering, browsing and ranking. Indexing is performed once off-line. The other three steps are performed iteratively during the search task. The aim of the *indexing* step is to provide users with a set of high-level entry points into the dataset. We use a set of 17 specific concept detectors developed by CMU for the TRECVID, such as female speech, aircraft and newsSubjectMonologue. We augment the high-level concepts by deriving textual concepts from the speech recognition result using Latent Semantic Indexing (LSI). Thus we decompose the information space into a small set of broad concepts, where the selection of one word from the concept reveals the complete set of associated words also.

For all keyframes in the dataset low-level indexing is performed by computing the global Lab color histograms. To structure these low-level visual descriptions of the dataset, the whole dataset is clustered using k-means clustering with random initialization. The k in the algorithm is set to 143 as this is the number of images the display will show to the user. In summary, the off-line indexing stage results in three types of metadata associated with each keyframe: (1) the presence or absence of 17 high-level concepts, (2) words occurring in the shot extended with associated words and (3) a color histogram.

After indexing, the interactive process starts. Users first *filter* the total corpus of video by using the indexing data. Two options are available for filtering: selecting a high-level concept, and entering a textual query that is used as a concept. These can be combined in an 'and' search, or added in an 'or' search.

The filtering stage leads to an active set of shots represented as keyframes, which are used in the next step, *browsing*. At this point in the process it is assumed that the user is going to select relevant keyframes from within the active set. To get an overview of the data the user can decide to look at the clustered data, rather than the whole dataset. In this visualization mode, the central keyframe of each cluster is presented on the screen, in such a way that the distances between keyframes are preserved as good as possible. The user interface does not play the shots as clips since too much time would be spent on viewing the video clips.

When the user has selected a set of suitable images, the user can perform a *ranking* through query-by-example using the color histograms with Euclidean distance. The closest matches within the filtered set of 2,000 shots are computed, where the system alternates between the different examples selected. The result is a ranked list of 1,000 keyframes.

## 4 Methods

We observed search behavior of students using the video-retrieval system described in Sect. 3. The study was done as an addition to a submission to TRECVID. Apart from the search results that were collected and submitted to TRECVID, additional user-related variables were collected.

For the TRECVID 24 topics had to be found in a dataset consisting of 60 hours of video from ABC, CNN and C-SPAN. 21 Groups of students (18 pairs and 3 individuals) were asked to search for 12 topics. The topics were divided into two sets of 12 (topics 1-12 and topics 13-24) and assigned a set to each student pair. For submissions to TRECVID the time to complete one topic was limited to 15 minutes. Prior to the study the students received a three-hour training on the system. Five types of data were recorded:

**Entry Questionnaire.** Prior to the study all participants filled in a questionnaire in which data was acquired about the subject pool: gender, age, subject of study, year of enrollment, experience with searching.

**Average Precision.** Average precision (AP) was used as the measure of quality of the results of a search. AP is the average of the precision value obtained after each relevant camera shot is encountered in the ranked list [1]. Note that AP is a quality measure for *one search* and not the mean quality of a group of searches. AP of each search was computed with a ground truth provided by TRECVID. Since average precision fluctuates during the search, we recorded not only the average precision at the end of the search but also the maximum average precision during the search.

**Logfiles.** Records of user actions on the system were made containing the following data about each search: textual queries, high-level features used, type of query ('and' or 'or'), number of images selected, duration of the search. These data were collected at two points in time: at the end of the search and at the point at which maximum average precision was reached. The logfile data are used to answer the first research question.

**Topic Questionnaire.** After each search the participants answered 5 questions about the search: 1. Are you familiar with this topic? 2. Was it easy to get started on this search? 3. Was it easy to do the search on this topic? 4. Are you satisfied with your search results? 5. Do you expect that the results of this search contain a lot of non-relevant items (low precision)? All questions were answered on a 5-point scale (1=not at all, 5=extremely). The resulting data were used as input for answering the second research question.

**Exit Questionnaire.** After the study all participants filled in a short questionnaire containing questions about the user's opinion of the system and the similarity between this type of search and the searches that they were used to perform.

To answer the third research question, the topics were categorized using a framework that was designed for a previous study [2]. The framework combines different methods to categorize image descriptions (e.g [3] and [4]) and divides queries into various levels and classes. For the present study we used only those distinctions that we considered relevant to the list of topics provided by TRECVID (Table 1): "general" vs. "specific" and "static" vs. "dynamic". Other distinctions, such as "object" vs. "scene", were not appropriate for the topic list since most topics contained descriptions of both topics and scenes.

**Table 1.** Summary of topics, categorized into general and specific and into dynamic and static. See <http://www.cs.vu.nl/~laurah/trec/topics.html> for topic details.

Class	General	Specific
Static	18: a crowd in urban environment 16: road with vehicles 14: snow-covered mountains 13: flames 01: aerial view of buildings 10: tank 22: cup of coffee 23: cats 06: helicopter	09: the mercedes logo 25: the white house 07: tomb of the unknown soldier 17: the sphinx 24: Pope John Paul II 04: Yassar Arafat 20: Morgan Freeman 15: Osama bin Laden 19: Mark Souder
Dynamic	05: airplane taking off 12: locomotive approaching you 08: rocket taking off 11: person diving into water	02: basketball passing down a hoop 03: view from behind catcher while pitcher is throwing the ball

## 5 Subjects

The subjects participating in the study were 39 students in Information Science who enrolled in the course Multimedia Retrieval at the University of Amsterdam. The number of years of enrollment at the university was between 1 and 8 (mean = 3.5). Two were female, 37 male. Ages were between 20 and 40 (mean=23.4).

Before the start of this study, we tested the prior search experience of the subjects in a questionnaire. All subjects answered questions about frequency of use and experience with information retrieval systems in general and, more specifically, with multimedia retrieval systems. It appeared that all students searched for information at least once a week and 92 % had been searching for two years or more. All students searched for multimedia at least once a year, and 65 % did this once a week or more. 88 % of the students had been searching for multimedia for at least two years. This was tested to make sure that prior search experience would not interfere with the effect of search strategies on the results. We did not find any evidence of a correlation between prior search experience and strategy, nor between prior search experience and search results. The lack of influence of search experience can in part be explained from the fact that the system was different from search systems that the students were used to. All but three students indicated in the exit questionnaire that the system was not at all similar to what they were used to. All students disagreed with or were neutral to the statement that the topics were similar to topics they typically search for. Another possible reason for the absence of an effect of prior search experience is the three-hour training that all students had received before the study.

The subjects indicated a high familiarity with the topics. Spearman's correlation test indicated a relationship between familiarity and average precision only within topics 10 and 13. We do not consider this enough evidence that there is in fact a relationship.

## 6 Results

The data were analyzed on the level of individual searches. A search is the process of one student pair going through the three interactive stages of the system for one topic. 21 Groups of students searched for 12 topics each, resulting in 252 searches. After exclusion of searches that were not finished, contained too much missing data, or exceeded the by TRECVID imposed maximum of 15 minutes, 242 searches remained.

**User actions.** In Table 2 descriptives are presented of the variables recorded in the logfiles. It shows that a search took approximately 8 minutes; 9 images were selected per search; high-level features were hardly used; or-search was used more than and-search.

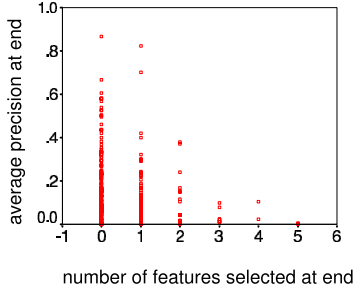
The mean average precision at the end of a search was 0.16. *Number of selected images* was the most important variable to explain the result of a search. This can be explained by the fact that each correctly selected image adds at least one relevant image to the result set. The contribution of the ranking to the result was almost negligibly small; change in AP caused by the ranking step had a mean of 0.001 and a standard deviation of 0.032. *Number of selected images* was not correlated to *time to finish topic*, *number of features*, or *type of search*.

There was no correlation between *time to finish topic* and average precision, nor between *type of search* and average precision. *Number of high-level features*

**Table 2.** User actions in the system at the moment of maximum AP and at the end of the search

	N	Max				End			
		Min.	Max.	Mean	St.D.	Min.	Max.	Mean	St.D.
Time (sec.)	242	0	852	345	195	6	899	477	203
No. of images selected	242	0	30	8.47	7.01	0	30	9.07	7.06
No. of high-level features	240	0	5	0.50	0.84	0	17	0.59	1.39
‘And’ or ‘Or’ search	240	And:75 Or:165				And:82 Or:158			

had a negative influence on the result. This is depicted in Fig. 1. The number of uses per features was too low to draw conclusions about the value of each feature. We can conclude, however, that selection of more than one feature leads to low average precision. To give an indication of the quality of the features that were used by the students, Table 3 shows the frequency of use and the mean average precision of the features. Only searches in which a single feature was used are included.



**Fig. 1.** Scatterplot of number of selected features and AP at the end of the search. One case with 17 features and AP of 0.027 is left out of the plot.

**User prediction of search quality.** In the topic questionnaire we collected opinions and expectations of users on a particular search. All questions measure an aspect of the user’s estimation of the search. For each question it holds that a high score represents a positive estimation, while a low score represents a negative estimation. Mutual dependencies between the questions complicate conclusions on the correlation between each question and the measured average precision of a search. Therefore, we combined the scores on the 4 questions into one variable, using principal component analysis. The new variable that is thus created represents the combined user estimation of a search. This variable explains 70 % of the variance between the cases. Table 4 shows the loading of

**Table 3.** High-level features: mean average precision and standard deviation.

Feature	N	Mean AP	St.d.	Feature	N	Mean AP	St.d.
Aircraft	5	0.09	0.05	People	3	0.13	0.15
Animal	5	0.17	0.06	PersonX	7	0.14	0.16
Building	2	0.30	0.00	PhysicalViolence	0	.	.
CarTruckBus	4	0.11	0.03	Road	3	0.06	0.04
FemaleSpeech	0	.	.	SportingEvent	9	0.08	0.03
NewsSubjectFace	1	0.24	.	Vegetation	1	0.13	.
NewsSubjectMonologue	1	0.70	.	WeatherNews	0	.	.
NonStudioSetting	4	0.15	0.13	ZoomIn	1	0.08	.
Outdoors	15	0.17	0.20				

each question on the first principal component. Pearson’s correlation test showed a relationship between combined user estimation and actually measured average precision. (Pearson’s correlation coefficient ( $P_{cc}$ ) = 0.298,  $\alpha = 0.01$ ). This suggests that users are indeed able to estimate the success of their search.

**Table 4.** Principal Component Analysis

Questionnaire item	Component 1
easy to start search	0.869
easy to do search	0.909
satisfied with search	0.874
expect high precision	0.678

Another measure of user estimation of a search is the difference between the point where maximum precision was reached and the point where the user stopped searching. As mentioned in Sect. 6, the mean time to finish a search was 477 seconds, while the mean time to reach maximum average precision was 345 seconds. The mean difference between the two points in time was 128 seconds, with a minimum of 0, a maximum of 704 and a standard deviation of 142 seconds. This means that students typically continued their search for more than two minutes after the optimal result was achieved. This suggests that even though students were able to estimate the overall success of a search, they did not know when the best results were achieved within a search. A correlation between combined user estimation and time-after-maximum-result shows that the extra time was largest in searches that got a low estimation ( $P_{cc} = -0.426$ ,  $\alpha = 0.01$ ). The extra 2 minutes did not do much damage to the precision. The mean average precision of the end result of a search was 0.16, while the mean maximum average precision of a search was 0.18. The mean difference between the two was 0.017, with a minimum of 0, a maximum of 0.48 and a standard deviation of 0.043.



**Topic type.** Table 5 shows that “specific” topics were better retrieved than “general” topics. The results of “static” topics were better than the results of “dynamic” topics. These differences were tested with an analysis of variance. The differences are significant far beyond the 0.01  $\alpha$ -level. We did not find any evidence that user actions were different in different categories.

**Table 5.** Mean AP of topics types, and ANOVA results

Mean AP	Static	Dynamic	Total	ANOVA results	SS	df	MS	F	Sig.
General	0.12	0.10	0.11	Between Groups	0.426	1	0.426	18.109	0.000
Specific	0.27	0.08	0.22	Within groups	5.648	240	0.024		
Total	0.19	0.10	0.16	Total	6.074	241			

The change in AP caused by the ranking step was positive for general topics (mean = 0.005), while negative for specific topics (mean = - 0.004). For general topics we found a correlation between *change in AP* and *AP at the end of the search* ( $P_{cc} = 0.265$ ,  $\alpha = 0.004$ ), which was absent for specific topics.

## 7 Discussion

Different types of topics result in differences in the quality of the search results. Results of “specific” topics were better than results of “general” topics. This suggests that indexing and filtering are the most important steps in the process. These steps are based on text retrieval, where it is relatively easy to find uniquely named objects, events or people. In content-based image retrieval on the other hand, and especially when the image is concerned as a whole, it is difficult to distinguish unique objects or people from other items of the same category. We are planning to upgrade the system so that regions within an image can be dealt with separately. Results of “static” topics were better than results of “dynamic” topics. This can be explained by the fact that the system treats the video data in terms of keyframes, *i.e.*, still images.

From the recorded user actions, *number of selected images* is by far the most important for the result. This is mainly caused by the addition of correctly selected images to the result set. The contribution of the ranking step to the average precision was almost negligibly small. We conclude from this that the main contribution of content-based image retrieval to the retrieval process is visualization of the dataset which gives the user the opportunity to manually select relevant keyframes. The visualization of the data set also gives the user an overview of the data and thus an indication of the success of the search. The results of the study show that users can estimate the success of a search quite well, but do not know when the optimal result is reached within a search.

This study reflects user behavior on one particular system. However, the results can to a certain extent be generalized to other interactive video-retrieval systems. The finding that “specific” topics are better retrieved than “general”

topics is reflected by the average TRECVID results. The fact that users do not know when to stop searching is a general problem of category search [5], where a user is searching for shots belonging to a certain category rather than for one specific shot. One solution to this problem is providing the user with an overview of the dataset. Future research is needed to compare the effectiveness of different types of visualization.

One of the reasons for this study was to learn which user variables are of importance for video retrieval, so that these variables can be measured in a future experiment. The most discriminating variable in the study proved to be the *number of selected images*. Further research is needed in which the optimal number of examples in a query-by-example is determined, taking in account the time spent by a user. In addition, future research is needed in which the four steps in the system are compared. In an experimental setting text-based retrieval and content-based retrieval can be compared. It would also be interesting to compare the results of an interactive video retrieval system to sequential scanning of shots in the data set for a fixed amount of time.

One of the results was that prior experience with searching and familiarity with the topic do not affect the quality of the search results. The latter seems to indicate that background knowledge of the searcher about the topic is not used in the search process. Some attempts to include background knowledge into the process of multimedia retrieval are made (see for example [6,7]). We would be interested to see how these techniques can be incorporated in an interactive video retrieval system.

## References

1. M.Worring, G.P.Nguyen, L.Hollink, J.Gemert, D.C.Koelma: Interactive search using indexing, filtering, browsing, and ranking. In: Proceedings of TRECVID. (2003)
2. L.Hollink, A.Th.Schreiber, Wielinga, B., M.Worring: Classification of user image descriptions. International Journal of Human Computer Studies (2004) To Appear.
3. Armitage, L., Enser, P.: Analysis of user need in image archives. Journal of Information Science **23** (1997) 287–299
4. Jorgensen, C.: Attributes of images in describing tasks. Information Processing and Management **34** (1998) 161–174
5. Smeulders, A., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000)
6. L.Hollink, A.Th.Schreiber, J.Wielemaker, B.Wielinga: Semantic annotation of image collections. In: Proceedings of the K-CAP 2003 Workshop on Knowledge Markup and Semantic Annotation, Florida, USA (2003)
7. A.Jaimes, B.L.Tseng, J.R.Smith: Modal keywords, ontologies, and reasoning for video understanding. In E.M.Bakker, ed.: CIVR. Volume 2728 of LNCS. (2003)

# Video Content Foraging

Ynze van Houten<sup>1</sup>, Jan Gerrit Schuurman<sup>1</sup>, and Pløn Verhagen<sup>2</sup>

<sup>1</sup>Telematica Instituut,

P.O.Box 589, 7500 AN Enschede, The Netherlands

{Ynze.vanHouten, JanGerrit.Schuurman}@telin.nl

<sup>2</sup>Educational Science and Technology, University of Twente,

P.O.Box 217, 7500 AE Enschede, The Netherlands

p.w.verhagen@utwente.nl

**Abstract.** With information systems, the real design problem is not increased access to information, but greater efficiency in finding useful information. In our approach to video content browsing, we try to match the browsing environment with human information processing structures by applying ideas from information foraging theory. In our prototype, video content is divided into video patches, which are collections of video fragments sharing a certain attribute. Browsing within a patch increases efficient interaction as other video content can be (temporarily) ignored. Links to other patches (“browsing cues”) are constantly provided, facilitating users to switch to other patches or to combine patches. When a browsing cue matches a user’s goals or interests, this cue carries a “scent” for that user. It is stated that people browse video material by following scent. The prototype is now sufficiently developed for subsequent research on this and other principles of information foraging theory.

## 1 Introduction

Humans are *informavores*: organisms that hunger for information about the world and about themselves [1]. The current trend is that more information is made more easily available to more people. However, a wealth of information creates a poverty of directed attention and a need to allocate sought-for information efficiently (Herbert Simon in [2]). The real design problem is not increased access to information, but greater efficiency in finding useful information. An important design objective should be the maximisation of the allocation of human attention to information that will be useful. On a 1997 CHI workshop on Navigation in Electronic Worlds, it was stated [3]: “Navigation is a situated task that frequently and rapidly alternates between discovery and plan-based problem-solving. As such, it is important to understand each of the components of the task – the navigator, the world that is navigated, and the content of that world, but equally important to understand the synergies between them.” Information foraging theory [4] can be an important provider of knowledge in this regard, as it describes the information environment and how people purposefully interact with that environment.

In this paper, we describe the design of a video-interaction environment based upon ideas from information foraging theory. The environment is developed to test these ideas in subsequent research, as will be explained at the end of this paper.

Finding video content for user-defined purposes is not an easy task: video is time-based, making interacting with video cumbersome and time-consuming. There is an urgent need to support the process of efficiently browsing video content. An orderly overview of existing video browsing applications and related issues can be found in [5]. Our approach adds a new perspective in that it applies a human-computer interaction theory to the problem of video content browsing.

Emphasis is on browsing - and not on querying - for a number of reasons. To start with, people are visual virtuosos [6]. In visual searching, humans are very good at rapidly finding patterns, recognising objects, generalising or inferring information from limited data, and making relevance decisions. The human visual system can process images more quickly than text. For instance, searching for a picture of a particular object is faster than searching for the *name* of that object among other words [7]. Given these visual abilities, for media with a strong visual component, users should be able to get quick access to the images. In the case of video, its richness and time-basedness can obstruct fast interaction with the images, so efficient filter and presentation techniques are required to get access to the images.

Except when the information need is well defined and easily articulated in a (keyword) query, browsing is an advantageous searching strategy because in many cases users do not know exactly what they are looking for. Well-defined search criteria often crystallise only in the process of browsing, or initial criteria are altered as new information becomes available. A great deal of information and context is obtained along the browsing path itself, not just at the final page. The search process itself is often as important as the results. Moreover, users can have difficulty with articulating their needs verbally, which especially applies in a multimedia environment, where certain criteria do not lend themselves well to keyword search. Furthermore, appropriate keywords for querying may not be available in the information source, and if they are available, the exact terminology can be unknown to the user [8].

Browsing is a search strategy closer related to “natural” human behaviour than querying [9]. As such, a theory describing natural behaviour in an information environment may be very useful when designing a video browsing environment. Information foraging theory addresses this topic.

## 2 Information Foraging Theory (IFT)

In this paper, we describe the design of a video browsing environment based upon ideas from information foraging theory [4]. IFT is a “human-information interaction” theory stating that people will try to interact with information in ways that maximise the gain of valuable information per unit cost. Core elements of the theory that we apply are:

- People forage through an information space in search of a piece of information that associates with their goals or interests like animals on the forage for food.
- For the user, the information environment has a “*patchy*” structure (compare patches of berries on berry bushes).
- Within a patch, a person can decide to forage the patch or switch to another patch.
- A strategy will be superior to another if it yields more useful information per unit cost (with cost typically measured in time and effort).

- Users make navigational decisions guided by *scent*, which is a function of the perception of value, cost, and access path of the information with respect to the goal and interest of the user.
- People adapt their scent-following strategies to the flux of information in the environment.

For applying IFT ideas to building an environment that supports efficient video browsing we need patches, and scent-providing links (browsing cues) to those patches. The patches provide structure to the information environment. Patches are expected to be most relevant when patches as defined in the database match with patches as the user would define them. To facilitate the use of patches, users need to be helped to make estimates of the gain they can expect from a specific information patch, and how much it will cost to discover and consume that information. These estimates are based on the user's experience, but also on the information provision of browsing cues.

The concept of scent provides directions to the design of information systems as it drives users' information-seeking behaviour. When people perceive no scent, they should be able to perform a random walk in order to spot a trace of scent. When people perceive a lot of scent, they should be able to follow the trail to the target. When the scent gets low, people should be able to switch to other patches. These types of browsing behaviours all need to be supported in the design. Typical design-related situations can be that browsing cues are misleading (the scent is high, but the target is not relevant/interesting) or badly presented (no or low scent, but a very relevant or interesting target).

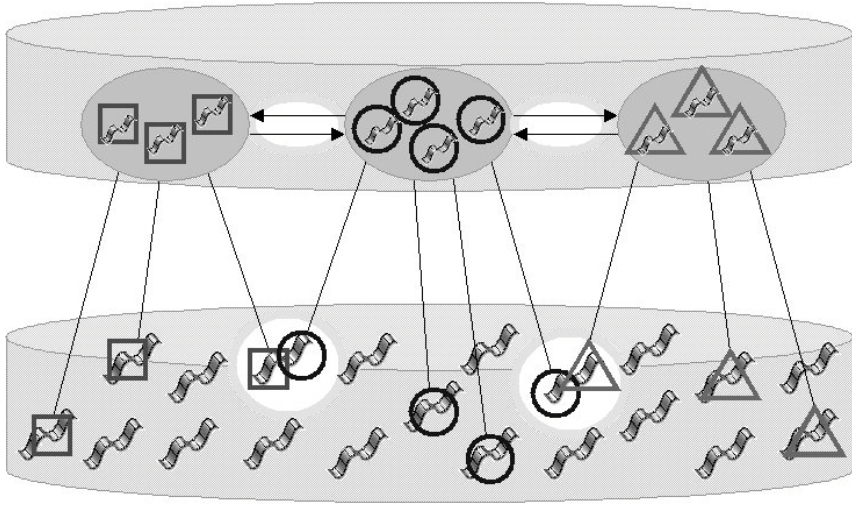
## 2.1 Video Patches

A video can be looked at as a database containing individual video fragments [10]. The original narrative of the video is “only” one out of many ways of organising and relating the individual items. People often want to structure the information environment in their own way, where the “decodings are likely to be different from the encoder's intended meaning” [11].

Video patches are collections of video fragments sharing a certain attribute (see Figure 1). Attributes may vary along many dimensions, including complex human concepts and low-level visual features. Patches can form a hierarchy, and several combinations of attributes can be combined in a patch.

As video fragments can have a number of attributes (e.g., a fragment can contain certain people, certain locations, certain events etc.), the fragments will occur in a number of patches. When viewing a fragment in a patch, links to other patches can be presented to the user. As such, patches form a hyperlinked network.

Patches provide easy mechanisms to filter video content as users can browse a patch and ignore video fragments not belonging to the patch (see also Figure 3). What are good patches depends on issues like the user's task and the video genre. User experiments are needed to establish which patches are useful in which situation.



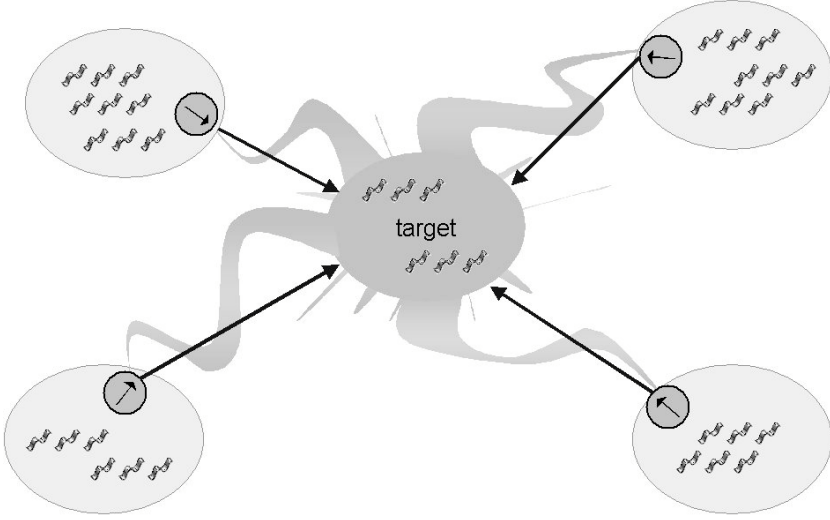
**Fig. 1.** Representation of video patches. The lower container is a database with video fragments (visualised as filmstrips), which can be the semantic units of a video. Fragments with the same attributes (here: squares, circles, triangles) are combined in video patches. The upper container is the browsing environment containing video patches (here: dark ellipses). When a fragment appears in two or more patches, links between these patches emerge (here: arrows between patches).

## 2.2 The Scent of a Video Patch

Certain video patches (or items within those patches) can have a semantic match with the user's goal or interests, and as such, give off scent. A user's goal or interest activates a set of chunks in a user's memory, and the perceived browsing cues leading to patches (or patch items) activate another set of chunks. When there is a match, these browsing cues will give off scent. Users can find the relevant patches by following the scent. Scent is wafted backward along hyperlinks – the reverse direction from browsing (see Figure 2). Scent can (or should) be perceivable in the links – or *browsing cues* – towards those targets. Users act on the browsing cues that they perceive as being most semantically similar to the content of their current goal or interest (see also [12]). For example, the scent of menus influences the decision whether users will use them or start a query instead [13].

In terms of IFT, The perception of information scent (via browsing cues) informs the decisions about which items to pursue so as to maximise the information diet of the forager. If the scent is sufficiently strong, the forager will be able to make the informed choice at each decision point. People switch when information scent gets low. If there is no scent, the forager will perform a random walk. What we are interested in is what exactly determines the amount of scent that is perceived.

We have built an experimental video-browsing application (Figure 3) to study a) what are good attributes to create patches with, and b) how to present browsing cues in such a way they can present scent to the user.

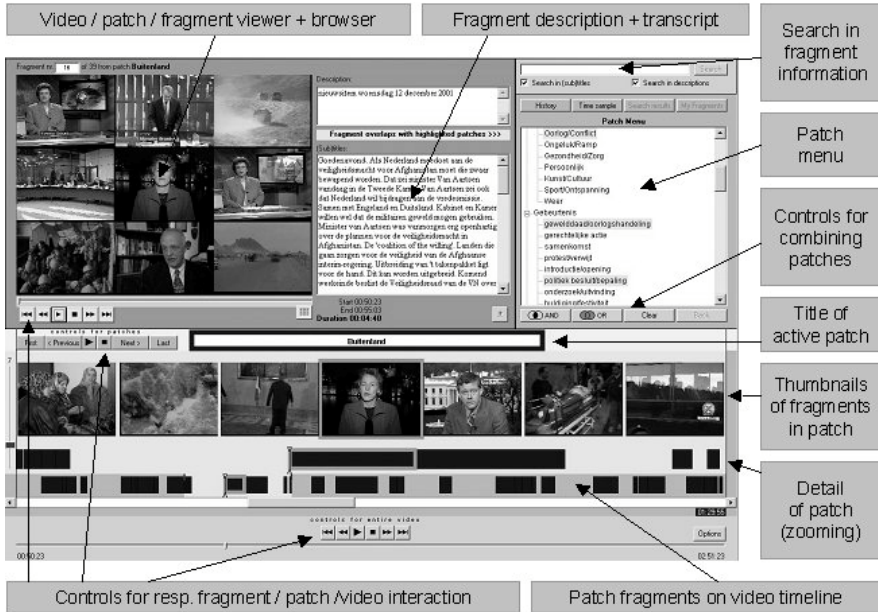


**Fig. 2.** Schematic illustration of the scent of a video patch. One or more video fragments in the central patch semantically matches with the user's goals or interests, and therefore can be considered a target. The browsing cues in surrounding patches that link to the target carry "scent", which is wafted backwards along the hyperlinks.

### 3 Design of a Video-Browsing Application

We designed a prototype of a video browsing tool applying the ideas from IFT (see Figure 3). The aim of the prototype is to validate the applicability of IFT for browser design. For that purpose, the video that is browsed needs to be prepared in advance as a set of patches and scent-carrying cues. The richness of semantic cues that have to be taken into account goes beyond the current state of the art in research about automated feature extraction, structure analysis, abstraction, and indexing of video content (see for instance [14], [15], [16], [17]). The automated processes that are technologically feasible cannot yet accomplish the video browsing structure that is cognitively desirable. For the experimental prototype we prepared the video mostly by hand.

In the prototype, scent-based video browsing will not be independent of querying. Smeaton [18] states that for video we need a browse-query-browse interaction. Querying helps to arrive at the level of video patches where the user needs to switch to browsing. Query-initiated browsing [19] is demonstrated to be very fruitful [20]. Golovchinsky [21] found that in a hypertext environment, (dynamic) query-mediated links are as effective as explicit queries, indicating that the difference between queries and links (as can be found – for example – in a menu) is not always significant. In the application described here, we see that the user can query the database and actually is browsing prefabricated queries.



**Fig. 3.** Interface of the patch-based video browsing application. Users can start interacting with the video by a) selecting a patch from the patch menu, b) querying video fragment information after which results are presented as a patch, or c) simply playing the video. When a patch is selected, the patch fragments - represented by boxes - are displayed on the video timeline, thus displaying frequency, duration, and distribution information of the fragments in the patch. To get more focussed information, A part of the patch is zoomed in on, and for these items keyframes are presented. Always one fragment in the patch is “active”. For this item, the top-left part of the interface presents 9 frames (when the item is not played or scrolled using the sidebar). All available information about the fragment is displayed (transcript, text on screen, descriptions). For the activated fragment it is shown in which other patches it appears by highlighting those patches in the patch menu.

### 3.1 Patch Creation and Presentation

When browsing starts and the user wants to select a first patch from the menu, the patch with the highest scent will probably get selected (“this is where I will probably find something relevant/of interest”). Which types of patches are created and the way they are presented (labelled) is here the central issue.

In order to create patches, the video is first segmented into semantically meaningful fragments, which will become the patch items in the video patches. What are relevant units is genre-dependent. For a newscast, the newsitems seem to be the best units. For an interview, it may be each new question or group of related questions. For a football game, it may be the time between pre-defined “major events” (e.g. goals, cards, etc.).

For each video unit, attributes are defined (in other words, what is *in* this fragment? What is this fragment about?). This step defines the type (or theme) of the video



patches, and as such, the main entries in the patch menu that will be formed. User inquiries are needed to find out what are the most useful attributes. For a football match, it may be players, goals, shots on goal, cards etc. For a newscast, this may be the news category, persons, locations, events, etc.

Next, attribute values need to be defined (so, not just whether there is a person, but also who *is* that person). This step defines the specific patches that will be used in the browsing process. The main question here is which values will be most useful, which will depend on contexts of use. This is a typical example why much is done by hand: automatic feature extraction techniques currently have great difficulties performing this task. Data from closed captions, speech recognition, and OCR (optical character recognition), however, proved to be most helpful.

Fragments sharing the same attribute values are combined into patches, and these patches are indexed. All this information about the video is stored in MPEG7 [22].

### 3.2 Support for Different Types of Browsing Behaviour

As noted earlier, we distinguish three types of browsing behaviour: a random walk, within-patch browsing, and switching (between-patches browsing).

For a random walk, users can play or browse the video as a whole, or browse the patch containing all video fragments (thus getting information for each fragment).

For within-patch browsing, users can simply play the patch “hands-free”: at the end of each fragment the video jumps to the start of the next fragment in the patch. Alternatively, users can move the zooming window (using a slidebar) to scan the thumbnails representing the fragments in the patch. By clicking a thumbnail (or the neutral boxes representing a fragment), the user “activates” a fragment, thus displaying a lot of detailed information about that fragment. Users can use “next” and “previous” buttons to easily see detailed information about other individual fragments. As such, users can quickly scan within a patch what is or is not relevant (that is, what does and does not associate with their goals or interests).

For every patch item, it is shown in which other patches it appears. Highlighting those patches in the patch menu indicates this. This also provides metadata about the current item. For example, when watching an item from the patch “politicians”, the item “drugs” may be highlighted in the menu, indicating what issue a politician is referring to. When the user is interested in video fragments on drugs, the user can simply switch to that patch. When the user is interested in opinions of politicians about drugs, the user may combine the two patches using the logical operator AND (or OR, if the user is interested in both). This way, users can influence the structure of the information environment in a way IFT calls “enrichment” [4]. Of course, users can always switch to any other patch in the patch menu.

When the user wants to switch, the interface needs to present possibilities (that is, browsing cues) to switch to other patches. Reasons people want to switch may occur when: a) the current source has a scent below a certain threshold, or b) the user is inspired by what is seen within the current source (“I want to see more like this!”), or c) the browsing cues simply give off a lot of scent (cues point directly to sought-for information). If necessary, browsing cues can be presented dynamically, that is, query- or profile-dependent (see also [19]).

### 3.3 Scent Presented in the Interface

When the user is browsing video data, the current patch or fragment will give off a certain scent via so-called “scent carriers”. Assuming that people are scent-followers, the main question here is: How can we present scent in such a way that users can efficiently and effectively browse video material?

The title of the patch is the first scent carrier the user is confronted with. When the patch name semantically matches the user’s goals or interest, it will carry a certain amount of scent. The way patch names are organised and presented will influence the amount of perceived scent.

When a patch is activated, several indicators will provide more or less scent, indicating to the user that “I’m on the right track”, or “I need to switch to another patch”. First of all, the frequency, duration, and distribution information of the fragments in the patch can provide a lot of scent (for example, when looking for the main people involved in a story, frequency information may be very useful). Still frames representing the fragments in the patch are the next scent carrying cues. For the one active fragment, a lot of scent carriers are available: keyframes (in this case, nine), transcript (derived from closed captions [when available] and/or speech recognition), displayed text in the video (optical character recognition), and added description. Of course, the video images— that can be either viewed or browsed by fast-forwarding or using a slider - can also carry scent by themselves.

Regarding switching to other patches, the way indications about overlapping patches are presented will influence the scent people perceive.

## 4 Conclusions and Future Work

The practical problem we try to deal with is how people can interact with video content in such a way that they can efficiently pursue their goals. We translate this to the research problem of how we can match the information environment with human information processing structures. Information foraging theory is assumed to be a fitting theory to answer this question as it both describes how people perceive and structure the information environment, and how people navigate through this environment. Our prototypical browsing environment is based on the principles of this theory. Hypotheses we derive from the theory is that humans perceive the information environment as “patchy”, humans navigate through the information environment by following scent, and humans will interact with the environment in ways that maximise the gain of valuable information per unit cost. We applied these ideas to construct a solution for efficient interaction with video content, as described in this paper. The actual development took place in a few steps applying an iterative design approach [23]. This is the starting point for our future research on the applicability of information foraging theory for the design of video interaction applications. As a first step we plan experiments to study the effectiveness of different presentations of browsing cues, how the perception of scent relates to different types of user tasks, how to support patch navigation, and which patches are useful in which situations.

## References

1. Miller, G. A. (1983). Informavores. In F. Machlup & U. Mansfield (Eds.), *The Study of Information: Interdisciplinary Messages* (pp. 111-113). New York: Wiley.
2. Varian, H. R. (1995). The Information Economy - How much will two bits be worth in the digital marketplace? *Scientific American*, September 1995, 200-201.
3. Jul, S. & Furnas, G. W. (1997). Navigation in electronic worlds: a CHI'97 workshop. *SIGCHI Bulletin*, 29, 44-49.
4. Pirolli, P. & Card, S. K. (1999). Information foraging. *Psychological Review*, 106, 643-675.
5. Lee, H. & Smeaton, A. F. (2002). Designing the user interface for the Físchlár Digital Video Library. *Journal of Digital Information*, 2.
6. Hoffman, D. D. (1998). *Visual intelligence*. New York, NY USA: W.W. Norton.
7. Paivio, A. (1974). Pictures and Words in Visual Search. *Memory & Cognition*, 2, 515-521.
8. Rice, R. E., McCreddie, M., & Chang, S.-J. L. (2001). *Accessing and browsing information and communication*. Cambridge, USA: MIT Press.
9. Marchionini, G. (1995). *Information seeking in electronic environments*. Cambridge University Press.
10. Manovich, L. (2001). *The language of new media*. Cambridge, MA: The MIT Press.
11. Hall, S. (1980). Encoding/Decoding. In *Culture, Media, Language: Working Papers in Cultural Studies 1972-1979*. London: Hutchinson.
12. Blackmon, M. H., Polson, P. G., Kitajima, M., & Lewis, C. (2002). Cognitive Walkthrough for the Web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '02* (pp. 463-470). Minneapolis, Minnesota, USA: ACM.
13. Katz, M. A. & Byrne, M. D. (2003). Effects of Scent and Breadth on Use of Site-Specific Search on E-commerce Web Sites. *ACM Transactions on Computer-Human Interaction*, 10, 198-220.
14. Yeo, B.-L. & Yeung, M. M. (1997). Retrieving and visualizing video. *Communications of the ACM*, 40, 43-52.
15. Dimitrova, N., Zhang, H. J., Shahraray, B., Sezan, I., Huang, T., & Zakhor, A. (2002). Applications of video-content analysis and retrieval. *IEEE MultiMedia*, 9(3), 42-55.
16. Wactlar, H. D. (2000). Informedia - search and summarization in the video medium. In *Proceedings of the Imagina 2000 Conference*.
17. Snoek, C. G. M., & Worring, M. (2002). A Review on Multimodal Video Indexing. In *Proceedings of the IEEE Conference on Multimedia & Expo (ICME)*.
18. Smeaton, A. F. (2001). Indexing, browsing and searching of digital video and digital audio information. In M. Agosti, F. Crestani, & G. Pasi (Eds.), *Lectures on information retrieval*. Springer Verlag.
19. Furnas, G. W. (1997). Effective view navigation. In *Proceedings of the conference on Human Factors in Computing Systems, CHI '97* (pp. 367-374). Atlanta, GA, USA: ACM.
20. Olston, C. & Chi, E. H. (2003). ScentTrails: Integrating Browsing and Searching on the Web. *ACM Transactions on Computer-Human Interaction*, 10, 177-197.
21. Golovchinsky, G. (1997). Queries? Links? Is there a difference? In *Proceedings of the Conference on Human Factors in Computing Systems, CHI '97* (pp. 407-414). Atlanta, GA, USA: ACM.
22. Nack, F. & Lindsay, A.T. (1999). Everything you wanted to know about MPEG-7: Part 1. *IEEE MultiMedia*, 6(3), 65-77.
23. van Houten, Y., van Setten, M., & Schuurman, J. G. (2003). Patch-based video browsing. In M. Rauterberg, M. Menozzi, & J. Wesson (Eds.), *Human-Computer Interaction INTERACT '03* (pp. 932-935). IOS Press.

# Using Maximum Entropy for Automatic Image Annotation

Jiwoon Jeon and R. Manmatha

Center for Intelligent Information Retrieval  
Computer Science Department  
University of Massachusetts Amherst  
{jeon, manmatha}@cs.umass.edu,

**Abstract.** In this paper, we propose the use of the Maximum Entropy approach for the task of automatic image annotation. Given labeled training data, Maximum Entropy is a statistical technique which allows one to predict the probability of a label given test data. The techniques allow for relationships between features to be effectively captured, and has been successfully applied to a number of language tasks including machine translation. In our case, we view the image annotation task as one where a training data set of images labeled with keywords is provided and we need to automatically label the test images with keywords. To do this, we first represent the image using a language of visterms and then predict the probability of seeing an English word given the set of visterms forming the image. Maximum Entropy allows us to compute the probability and in addition allows for the relationships between visterms to be incorporated. The experimental results show that Maximum Entropy outperforms one of the classical translation models that has been applied to this task and the Cross Media Relevance Model. Since the Maximum Entropy model allows for the use of a large number of predicates to possibly increase performance even further, Maximum Entropy model is a promising model for the task of automatic image annotation.

## 1 Introduction

The importance of automatic image annotation has been increasing with the growth of the worldwide web. Finding relevant digital images from the web and other large size databases is not a trivial task because many of these images do not have annotations. Systems using non-textual queries like color and texture have been proposed but many users find it hard to represent their information needs using abstract image features. Many users prefer textual queries and automatic annotation is a way of solving this problem.

Recently, a number of researchers [2,4,7,9,12] have applied various statistical techniques to relate words and images. Duygulu *et al.* [7] proposed that the image annotation task can be thought of as similar to the machine translation problem and applied one of the classical IBM translation models [5] to this problem. Jeon *et al.* [9] showed that relevance models (first proposed for information

retrieval and cross-lingual information retrieval [10]) could be used for image annotation and they reported much better results than [7]. Berger *et al.* [3] showed how Maximum Entropy could be used for the machine translation tasks and demonstrated that it outperformed the classical (IBM) machine translation models for the English-French translation task. The Maximum Entropy approach has also been applied successfully to a number of other language tasks [3,14].

Here, we apply Maximum Entropy to the same dataset used in [7,9] and show that it outperforms both those models. We first compute an image dictionary of visterms which is obtained by first partitioning each image into rectangular regions and then clustering image regions across the training set. Given a training set of images and keywords, we then define unigram predicates which pair image regions and labels. We automatically learn using the training set how to weight the different terms so that we can predict the probability of a label (word) given a region from a test image. To allow for relationships between regions we define bigram predicates. In principle this could be extended to arbitrary n-grams but for computational reasons we restrict ourselves to unigram and bigram predicates in this paper.

Maximum Entropy maximizes entropy i.e. it prefers a uniform distribution when no information is available. Additionally, the approach automatically weights features (predicates). The relationship between neighboring regions is very important in images and Maximum Entropy can account for this in a natural way.

The remainder of the paper is organized as follows. Related work is discussed in section 2. Sections 3 provides a brief description of the features and image vocabulary used while the Maximum Entropy model and its application to image annotation are briefly discussed in 4 Experiments and results are discussed in 5 while Section 6 concludes the paper.

## 2 Related Work

In image annotation one seeks to annotate an image with its contents. Unlike more traditional object recognition techniques [1,8,15,17] we are not interested in specifying the exact position of each object in the image. Thus, in image annotation, one would attach the label “car” to the image without explicitly specifying its location in the picture. For most retrieval tasks, it is sufficient to do annotation. Object detection systems usually seek to find a specific foreground object, for example, a car or a face. This is usually done by making separate training and test runs for each object. During training positive and negative examples of the particular object in question are presented. However, in the annotation scheme here background objects are also important and we have to handle at least a few hundred different object types at the same time. The model presented here learns all the annotation words at the same time. Object recognition and image annotation are both very challenging tasks.

Recently, a number of models have been proposed for image annotation [2, 4,7,9,12]. Duygulu *et al* [7] described images using a vocabulary of blobs. First,

regions are created using a segmentation algorithm like normalized cuts. For each region, features are computed and then blobs are generated by clustering the image features for these regions across images. Each image is generated by using a certain number of these blobs. Their *Translation Model* applies one of the classical statistical machine translation models to translate from the set of blobs forming an image to the set of keywords of an image. Jeon *et al.* [9] instead assumed that this could be viewed as analogous to the cross-lingual retrieval problem and used a *Cross Media Relevance Model* (CMRM) to perform both image annotation and ranked retrieval. They showed that the performance of the model on the same dataset was considerably better than the models proposed by Duygulu *et al.* [7] and Mori *et al.* [11].

The above models use a discrete image vocabulary. A couple of other models use the actual (continuous) features computed over each image region. This tends to give improved results. *Correlation LDA* proposed by Blei and Jordan [4] extends the Latent Dirichlet Allocation (LDA) Model to words and images. This model assumes that a Dirichlet distribution can be used to generate a mixture of latent factors. This mixture of latent factors is then used to generate words and regions. Expectation-Maximization is used to estimate this model. Lavrenko *et al.* proposed the *Continuous Relevance Model* (CRM) to extend the *Cross Media Relevance Model* (CMRM) [9] to directly use continuous valued image features. This approach avoids the clustering stage in CMRM. They showed that the performance of the model on the same dataset was a lot better than other models proposed.

In this paper, we create a discrete image vocabulary similar to that used in Duygulu *et al.* [7] and Jeon *et al.* [9]. The main difference is that the initial regions we use are rectangular and generated by partitioning the image into a grid rather than using a segmentation algorithm. We find that this improves performance (see also [6]). Features are computed over these rectangular regions and then the regions are clustered across images. We call these clusters visterms (visual terms) to acknowledge that they are similar to terms in language.

Berger *et al.* [3] proposed the use of Maximum Entropy approaches for various Natural Language Processing tasks in the mid 1990's and after that many researchers have applied this successfully to a number of other tasks. The Maximum Entropy approach has not been much used in computer vision or imaging applications. In particular, we believe this is the first application of the Maximum Entropy approach to image annotation

### 3 Visual Representation

An important question is how can one create visterms. In other words, how does one represent every image in the collection using a subset of items from a finite set of items. An intuitive answer to this question is to segment the image into regions, cluster similar regions and then use the clusters as a vocabulary. The hope is that this will produce semantic regions and hence a good vocabulary. In

general, image segmentation is a very fragile and erroneous process and so the results are usually not very good.

Barnard and Forsyth[2] and Duygulu *et al.* [7] used general purpose segmentation algorithms like Normalized-cuts[16] to extract regions. In this paper, we use a partition of the image into rectangular grids rather than a segmentation of the image. The annotation algorithm works better when the image is partitioned into a regular grid. than if a segmentation algorithm is used to break up the image into regions (see also [6]). This means that the current state of the art segmentation algorithms are still not good enough to extract regions corresponding to semantic entities. The Maximum Entropy algorithm cannot undo the hard decisions made by the segmentation algorithm. These segmentation algorithms make decisions based on a single image. By using a finer grid, the Maximum Entropy algorithm automatically makes the appropriate associations.

For each segmented region, we compute a feature vector that contains visual information of the region such as color, texture, position and shape. We used K-means to quantize these feature vectors and generate visterms. Each visterm represent a cluster of feature vectors. As in Duygulu et al [7] we arbitrarily choose the value of k. In the future, we need a systematic way of choosing the value.

After the quantization, each image in the training set can now be represented as a set of visterms. Given a new test image, it can be segmented into regions and region features can be computed. For each region, the visterm which is closest to it in cluster space is assigned.

## 4 Maximum Entropy for Image Annotation

We assume that there is a random process that given an image as an observation generates a label  $y$ , an element of a finite set  $Y$ . Our goal is to create a stochastic model for this random process. We construct a training dataset by observing the behavior of the random process. The training dataset consists of pairs  $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$  where  $x_i$  represents an image and  $y_i$  is a label. If an image has multiple labels,  $x_i$  may be part of multiple pairings with other labels in the training dataset. Each image  $x_i$  is represented by a vector of visterms. Since we are using rectangular grids, for each position of the cell there is a corresponding visterm.

### 4.1 Predicate Functions and Constraints

We can extract statistics from the training samples and these observations should match the output of our stochastic model. In Maximum Entropy, any statistic is represented by the expected value of a feature function. To avoid confusion with image features, from now on, we will refer to the feature functions as predicates. Two different types of predicates are used.

- **Unigram Predicate**

This type of predicate captures the co-occurrence statistics of a visual term

and a label. The following is an example unigram predicate that checks the co-occurrence of the label ‘tiger’ and the visterm  $v_1$  in image  $x$ .

$$f_{v_1, tiger}(x, y) = \begin{cases} 1 & \text{if } y = tiger \text{ and } v_1 \in x \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

If image  $x$  contains visual term  $v_1$  and has ‘tiger’ as a label, then the value of the predicate is 1, otherwise 0. We have unigram predicates for every label and visterm pair that occurs in the training data. Since, we have 125 visual terms and 374 labels, the total number of possible unigram predicates is 46750.

### – Bigram Predicate

The bigram predicate captures the co-occurrence statistic of two visterms and a label. This predicate attempts to capture the configuration of the image and the positional relationship between the two visterms is important. Two neighboring visterms are horizontally connected if they are next to each other and their row coordinates are the same. They are vertically connected if they are next to each other and their column coordinates are the same. The following example of a bigram predicate models the co-occurrence of the label ‘tiger’ and the two horizontally connected visterms  $v_1$  and  $v_2$  in image  $x$ .

$$f_{horizontal\_v_1 v_2, tiger}(x, y) = \begin{cases} 1 & \text{if } y = tiger \text{ and } x \text{ contains} \\ & \text{horizontally connected } v_1, v_2 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

If  $x$  contains horizontally connected visterms  $v_1$  and  $v_2$  and ‘tiger’ is a label of  $x$ , then the value of the predicate is 1. We also use predicates that captures the occurrence of two vertically connected visterms. In the same way, we can design predicates that use 3 or more visterms or more complicated positional relationships. However, moving to trigrams or even n-grams leads to a large increase in the number of predicates and the number of parameters that must be computed and this requires substantially more computational resources.

The expected value of a predicate with respect to the training data is defined as follow,

$$\tilde{p}(f) = \sum_{x, y} \tilde{p}(x, y) f(x, y) \quad (3)$$

where  $\tilde{p}(x, y)$  is a empirical probability distribution that can be easily calculated from the training data. The expected value of the predicate with respect to the output of the stochastic model should be equal to the expected value measured from the training data.

$$\sum_{x, y} \tilde{p}(x, y) f(x, y) = \sum_{x, y} \tilde{p}(x) p(y|x) f(x, y) \quad (4)$$



where  $p(y|x)$  is the stochastic model that we want to construct. We call equation 4 a constraint. We have to choose a model that satisfies this constraint for all predicates.

## 4.2 Parameter Estimation and Image Annotation

In theory, there are an infinite number of models that satisfy the constraints explained in section 4.1. In Maximum Entropy, we choose the model that has maximum conditional entropy

$$H(p) = - \sum_{x,y} \tilde{p}(x)p(y|x) \log p(y|x) \quad (5)$$

The constrained optimization problem is to find the  $p$  which maximizes  $H(p)$  given the constraints in equation 4. Following Berger et al [3] we can do this using Lagrange multipliers. For each predicate,  $f_i$ , we introduce a Lagrange multiplier  $\lambda_i$ . We then need to maximize

$$A(p, \lambda) = H(p) + \sum_i \lambda_i (p(f_i) - \tilde{p}(f_i)) \quad (6)$$

Given fixed  $\lambda$ , the solution may be obtained by maximizing  $p$ . This leads to the following equations [3]:

$$p(y|x) = \frac{1}{Z(x)} \exp \left[ \sum_{i=1}^k \lambda_i f_i(x, y) \right] \quad (7)$$

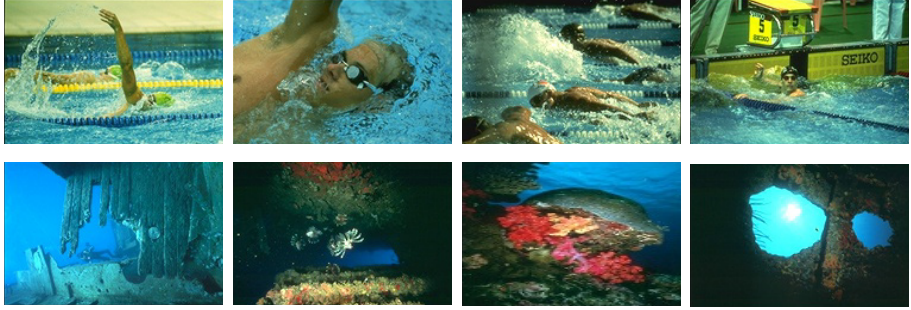
$$\Psi(\lambda) = - \sum_x \tilde{p}(x) \log Z(x) + \sum_i \lambda_i \tilde{p}(f_i) \quad (8)$$

where  $Z(x)$  is a normalization factor which is obtained by setting  $\sum_y p(y|x) = 1$ .

The solution to this problem is obtained by iteratively solving both these equations. A few algorithms have been proposed in the literature including Generalized Iterative Scaling and Improved Iterative Scaling [13]. We use *Limited Memory Variable Metric* method which is very effective for Maximum Entropy parameter estimation [13]. We use Zhang Le's [18] maximum entropy toolkit for the experiments in this paper.

**Table 1.** Experimental results

Experiment	recall	precision	F-measure
<i>Translation</i>	0.04	0.06	0.05
CMRM	0.09	0.10	0.10
Binary Unigram	0.11	0.08	0.10
Binary Unigram + Binary Bigram	0.12	0.09	0.11



**Fig.1.** Examples: Images in the first row are the top 4 images retrieved by query ‘swimmer’. Images in the second row are the top 4 images retrieved by query ‘ocean’.

## 5 Experiment

### 5.1 Dataset

We use the dataset in Duygulu *et al.*[7] to compare the models. We partition images into  $5 \times 5$  rectangular grids and for each region, extract a feature vector. The feature vector consists of average LAB color and the output of the Gabor filters. By clustering the feature vectors across the training images, we get 125 visterms.

The dataset has 5,000 images from 50 Corel Stock Photo cds. Each cd includes 100 images on the same topic. 4,500 images are used for training and 500 are used for test. Each image is assigned from 1 to 5 keywords. Overall there are 374 words (see [7]).

### 5.2 Results

We automatically annotate each test image using the top 5 words and then simulate image retrieval tasks using all possible one word queries. We calculate the mean of the precisions and recalls for each query and also the F-measure by combining the two measures using  $F = 1/(\lambda \frac{1}{P} + (1 - \lambda) \frac{1}{R})$  where P is the mean precision, R is the mean recall. We set the  $\lambda$  as 0.5.

In this paper, we used the results of the *Translation Model* [7] and the CMRM[9] as our baseline since they also use similar features. The experiment shows that Maximum Entropy using unigram predicates has performance comparable to the CMRM model (both have F-measures of 0.1). While one has better recall, the other has better precision. Both models outperform the classical translation model used by [7]. Using unigram and bigram predicates improves the performance of the Maximum Entropy model. Our belief is that by using predicates which provide even more configuration information, the model’s performance can be further improved.

Models which use continuous features (for example [12]) perform even better. Maximum Entropy models can also be used to model such continuous features

and future work will involve using such features. The results show that Maximum Entropy models have great potential and also enable us to incorporate arbitrary configuration information in a natural way.

## 6 Conclusions and Future Work

In this paper we show that Maximum Entropy can be used for the image annotation task and the experimental results show the potential of the approach. Since, we can easily add new types of predicate ( this is the one of the nice properties in Maximum Entropy ), there is great potential for further improvements in performance. More work on continuous valued predicates, image segmentation techniques and feature extraction methods will also lead to performance improvements.

**Acknowledgments.** We thank Kobus Barnard for making their dataset [7] available. We also thank Zhang Le [18] for making his Maximum Entropy Modeling Toolkit publicly available. This work was supported in part by the Center for Intelligent Information Retrieval and in part by grant #NSF IIS-9909073.

## References

1. S. Agarwal and D. Roth. Learning a Sparse Representation for Object Detection, IN *Proc. ECCV*, pages 113-130, 2002.
2. K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107-1135, 2003.
3. A. Berger, S. Pietra and V. Pietra. A Maximum Entropy Approach to Natural Language Processing. In *Computational Linguistics*, pages 39-71, 1996
4. D. Blei, and M. I. Jordan. Modeling annotated data. In *Proceedings of the 26th Intl. ACM SIGIR Conf.*, pages 127-134, 2003
5. P. Brown, S. D. Pietra, V. D. Pietra, and R. Mercer. The mathematics of statistical machine translation: Parameter estimation. In *Computational Linguistics*, 19(2):263-311, 1993.
6. P. Carbonetto, N. de Freitas. Why can't José read? The problem of learning semantic associations in a robot environment. In *Human Language Technology Conference Workshop on Learning Word Meaning from Non-Linguistic Data*, 2003.
7. P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Seventh European Conf. on Computer Vision*, pages 97-112, 2002.
8. R. Fergus, P. Perona and A. Zisserman. Object Class Recognition by Unsupervised Scale-Invariant Learning. IN *Proc. CVPR'03*, vol II pages 264-271, 2003.
9. J. Jeon, V. Lavrenko and R. Manmatha. (2003) Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. In *Proceedings of the 26th Intl. ACM SIGIR Conf.*, pages 119-126, 2003
10. V. Lavrenko, M. Choquette, and W. Croft. Cross-lingual relevance models. *Proceedings of the 25th Intl. ACM SIGIR Conf.*, pages 175-182, 2002.

11. Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *MISRM'99 First Intl. Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999.
12. V. Lavrenko, R. Manmatha and J. Jeon. A Model for Learning the Semantics of Pictures. In *Proceedings of the 16th Annual Conference on Neural Information Processing Systems*, 2004.
13. Robert Malouf. A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of the Sixth Workshop on Computational Language Learning*, 2003
14. K. Nigam, J. Lafferty and A. McCallum Using Maximum Entropy for Text Classification In *IJCAI-99 Workshop on Machine Learning for Information Filtering*, pages 61-67, 1999
15. H. Schneiderman, T. Kanade. A Statistical Method for 3D Object Detection Applied to Faces and Cars. *Proc. IEEE CVPR 2000*: 1746-1759
16. J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
17. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. CVPR'01*, pages 511-518, 2001.
18. Zhang Le. Maximum Entropy Modeling Toolkit for Python and C++. <http://www.nlplab.cn/zhangle/>

# Everything Gets Better All the Time, Apart from the Amount of Data

Hieu T. Nguyen and Arnold Smeulders

Intelligent Sensory Information Systems  
University of Amsterdam, Faculty of Science  
Kruislaan 403, NL-1098 SJ, Amsterdam, The Netherlands  
{tat,smeulders}@science.uva.nl

**Abstract.** The paper first addresses the main issues in current content-based image retrieval to conclude that the largest factors of innovations are found in the large size of the datasets, the ability to segment an image softly, the interactive specification of the user's wish, the sharpness and invariant capabilities of features, and the machine learning of concepts. Among these everything gets better every year apart from the need for annotation which gets worse with every increase in the dataset size. Therefore, we direct our attention to the question what fraction of images needs to be labeled to get an almost similar result compared to the case when all images would have been labeled by annotation? And, how can we design an interactive annotation scheme where we put up for annotation those images which are most informative in the definition of the concept (boundaries)? It appears that we have developed an random followed by a sequential annotation scheme which requires annotating 1% equal to 25 items in a dataset of 2500 faces and non-faces to yield an almost identical boundary of the face-concept compared to the situation where all images would have been labeled. This approach for this dataset has reduced the effort of annotation by 99%.

## 1 Introduction

With the progress in content-based image retrieval, we have left behind the early years [12]. Some patterns in the development and use of image and video retrieval systems are visible.

We give a short historical discussion on critical trends in the state of the art of content-based image retrieval and related topics:

No longer papers on computer vision methods deal with a *dataset size* of 50, as was common in the nineties, but typically datasets of thousands of images are being used today. The data are no longer standardized during recording, nor are they precisely documented. In contrast, large datasets have brought general datasets, which are weakly documented and roughly standardized. As an immediate consequence of the amount, analysis software has to be robust. And, the sheer number of data quickly favors methods and algorithms that are robust against many different sources of variation.

Another essential break through came when precise segmentation is unreachable and pointless for many tasks. *Weak segmentation* [12] strives towards finding a part of the object. To identify a scene, recognition of some details may suffice as long as the

details are unique in the database. Similarly, spatial relations between parts may suffice in recognition as long as the spatial pattern is unique.

*Interaction* is another essential step forward in useful content-based retrieval. Many users have not yet completely made up their minds what they want from the retrieval system. Even if they do, they surely want to grasp the result in a context also presenting images of similar relevance. The common scenario of interaction in full breadth is described in [19]. A different point of view is to perceive relevance feedback as a method of instant machine learning. In this paper we lay the foundation for the later issue.

Computer vision of almost any kind starts with features. Good features are capable of describing the semantics of relevant issues amidst a large variety of expected scenes and ignoring the irrelevant circumstances of the recording. Where the 80s and 90s concentrated on good features specific for each application, we now see a trend towards general and *complete sets of features invariant* to the accidental conditions of the recording while maintaining enough robustness and discriminatory power to distinguish among different semantics. Invariance classes can be of salient-photometric nature [11], color-photometric nature [3,4], given by occlusion and clutter requiring local features, geometrically [16,8]. In a recent comparison [11] between the various features set on the effectiveness, the SIFT set came out as the best one [7]. Robust, effective, narrow-invariant features are important as they focus maximally on the object-proper characteristics.

The application of *machine learning techniques* takes away the incidental variations of the concept and searches for the common at the level of the object class. Many different achievements have been made in object detection using tools of machine learning including the Support Vector Machines [9], boosting [18] and Gaussian mixture models [13].

All these items show progression every year. Segmentation has evolved into statistical methods of grouping local evidence. Interaction has absorbed relevance feedback, non-linear mapping on the screen and adaptable similarity measures. Features get more precise, more specifically invariant and powerful as well as more robust. Machine learning requires less data and performs better even if the boundary is very complex.

The foregoing shows that everything gets better all the time, apart from the amount of data. More data demand more effort in annotation up to the point where the data set gets so big that annotation is no longer feasible. Annotating thousands and eventually hundreds of thousands of pictures is hard to achieve, let alone the usual questions about the usefulness for one purpose and the reliability of such annotations. Where the machine power to make increasingly many computations is in place, the manpower for annotation will become the bottleneck.

Therefore, in this paper we concentrate on the following questions:

1. What fraction of images needs to be annotated with a label (while the rest is used unlabeled) to arrive at an almost similar definition (of the boundary) of the concept in feature space compared to the case when all images would have been labeled?
2. How to design a sequential selection rule yielding the most informative set of images offered to the user for annotation?
3. Can concepts be learned interactively in CBIR?

In section 2 the optimal data selection mechanism is discussed in the framework of active learning. The section points out the advantage of the discriminative probabilistic models for active learning. In particular, we present two well-founded methods for data selection. The proposed concepts are illustrated in section 3 which shows the results of each of the methods for annotation of human face images in a database.

## 2 Data Annotation Using Active Learning

Consider the annotation of the images in a database:  $\mathbf{x}_1, \dots, \mathbf{x}_N \in \mathbb{R}^d$  that is restricted to two classes with the labels  $+1, -1$  meaning “interesting” and “not interesting”, or “object of a certain type” and “not such object” or whatever other dichotomy. In effect we study the case of a one class classifier where we want to learn one specific concept. If there are more concepts to learn, it has to be done in sequel.

As labeling the entire database is considered impossible or at least inefficient, our aim is to label only a fraction of the images in the database denoted by  $\mathcal{D}_\ell$ . The remainder will stay unlabeled  $\mathcal{D}_u$ . We will rely on an existing classifier indicated by  $y = \text{sign}f(\mathbf{x})$  which will perform the assignment of all images in the database to the class or to the non-class.

The classifier needs to be trained on  $\mathcal{D}_\ell$ , and its performance varies for different labeled sets. This poses the question how to collect the optimal  $\mathcal{D}_\ell$  in term of the minimum number of images to label in order to arrive at a faithful best estimate of the concept boundary and the classification error. The problem is known as *active learning* [6]. It has been shown that a selective approach can achieve much better classifier than random selection.

Typically, data are collected one by one. Intuitively, samples in  $\mathcal{D}_u$  closest to the current classification boundary should be selected as they promise a large move of the boundary. We name this approach closest-to-boundary criterion. This heuristic approach has been used in active learning systems [6,14]. Other more theoretical criteria, however, proved to be inefficient [15,1]. Explaining this phenomenon, we notice that most existing methods use Support Vector Machines (SVM) [17] as the base learning algorithm. This classifier exhibits good performance in many pattern recognition applications, and works better than the generative methods for small and not randomly selected training sets. However, being an optimization with inequality constraints, SVM is inefficient for the common task in active learning when one needs to predict the classifier parameters upon the arrival of new training data. We argue that discriminative models with explicit probabilistic formulation are better models for active learning. The most popular is regularized logistic regression [21]. Other alternatives also exist [20]. Training such models boils down to the estimation of the parameters of the class posterior probability  $p(y|\mathbf{x})$  which is assumed to have a known form. While the discriminative probabilistic models have similar performance as SVM, they need only unconstrained optimization, and furthermore, are suitable for probabilistic analysis. The later advantages make the models more flexible in meeting sophisticated criteria in data selection.

The section presents two new active learning methods for the discriminative probabilistic models. They are better founded and has better performance than the closest-to-boundary method.

## 2.1 Data Selection by the Maximal Increase in Likelihood

In the first example, we consider the selection method that increases the likelihood of the labeled set as much as possible.

Suppose  $\theta$  is the parameter vector of the distribution  $p(y|\mathbf{x})$ . In the case of logistic regression:

$$p(y|\mathbf{x}) = \frac{1}{1 + \exp\{-y(\mathbf{a} \cdot \mathbf{x} + b)\}} \quad (1)$$

and  $\theta = \{\mathbf{a}, b\}$ ,  $\mathbf{a} \in \mathbb{R}^d, b \in \mathbb{R}$ . The parameters should be estimated via the minimization (maximization) of the regularized likelihood:

$$\mathcal{L} = -\ln p(\theta) - \sum_{i \in \mathcal{I}_\ell} \ln p(\mathbf{x}_i, y_i) - \sum_{i \in \mathcal{I}_u} \ln p(\mathbf{x}_i) \quad (2)$$

where  $\mathcal{I}_\ell$  and  $\mathcal{I}_u$  denote the set of indices of labeled and unlabeled samples respectively. The prior probability  $p(\theta)$ , usually defined as  $\exp\{-\frac{\lambda}{2} \|\mathbf{a}\|^2\}$ , is added as a regularization term to overcome numerical instability. The likelihood (2) can also be written as the sum of the likelihood of the class label:

$$\mathcal{L}^{(1)} = -\ln p(\theta) - \sum_{i \in \mathcal{I}_\ell} \ln p(\mathbf{x}_i | y_i; \theta) \quad (3)$$

and the likelihood of the data alone:

$$\mathcal{L}^{(2)} = - \sum_{i \in \mathcal{I}_\ell \cup \mathcal{I}_u} \ln p(\mathbf{x}_i) \quad (4)$$

Only  $\mathcal{L}^{(1)}$  needs to be minimized since  $\mathcal{L}^{(2)}$  does not depend on  $\theta$ . Let  $\hat{\theta}$  be the solution of the minimization of  $\mathcal{L}^{(1)}$  and  $\hat{\mathcal{L}}$  be the value of the minimum. Observe that  $\hat{\mathcal{L}}$  can only be increased when new data are labeled. This motivates the approach that selects the sample that maximizes the increase of  $\hat{\mathcal{L}}$  will promote a quicker convergence of the learning. Moreover, such a criterion will result in a training set with high likelihood, which would produce more reliable parameter estimates.

The increase of  $\hat{\mathcal{L}}$  when a prospective image  $\mathbf{x}_s$  is added to the labeled set, can be approximated analytically by the first step in Newton's minimization method without redoing the complete minimization:

$$\Delta \mathcal{L}_s \approx -\ln p(y_s | \mathbf{x}_s; \hat{\theta}) - \frac{1}{2} \nabla \ln p(y_s | \mathbf{x}_s)^T [\nabla^2 \mathcal{L} - \nabla^2 \ln p(y_s | \mathbf{x}_s)]^{-1} \nabla \ln p(y_s | \mathbf{x}_s) \quad (5)$$

The selection criterion can be based on the expectation of  $\Delta \mathcal{L}_s$  with respect to  $y_s$ :

$$s = \arg \max_{s \in \mathcal{I}_u} E_{y_s} [\Delta \mathcal{L}_s | \mathbf{x}_s] \quad (6)$$

$$E_{y_s} [\Delta \mathcal{L}_s | \mathbf{x}_s] = p(y_s = 1 | \mathbf{x}_s) \Delta \mathcal{L}_s^+ + p(y_s = -1 | \mathbf{x}_s) \Delta \mathcal{L}_s^- \quad (7)$$

where  $\Delta \mathcal{L}_s^+$  and  $\Delta \mathcal{L}_s^-$  denote the values of  $\Delta \mathcal{L}_s$  for  $y_s = 1$  and  $y_s = -1$  respectively. The probability  $p(y_s | \mathbf{x}_s)$  can be approximated using the current estimate  $\hat{\theta}$ . Note that one could also use  $\min\{\Delta \mathcal{L}_s^+, \Delta \mathcal{L}_s^-\}$  in place of the expectation without making an assumption on the accuracy of  $\hat{\theta}$ . In our current experiments, this criterion performs the same as eq. (6). Eq. (6) is therefore preferred as it is more attractive analytically.



## 2.2 Active Learning Using Pre-clustering and Prior Knowledge

In this section, we show another active learning method where the discriminative probabilistic models are used to take the advantage of pre-clustering and prior data knowledge.

In [2], Cohn et al. suggests a general approach to select the samples that minimize the expected future classification error:

$$\int_{\mathbf{x}} E [(\hat{y}(\mathbf{x}) - y(\mathbf{x}))^2 | \mathbf{x}] p(\mathbf{x}) d\mathbf{x} \quad (8)$$

where  $y(\mathbf{x})$  is the true label of  $\mathbf{x}$  and  $\hat{y}(\mathbf{x})$  is the classifier output. Although the approach is well founded statistically, the direct implementation is usually difficult since the data distribution  $p(\mathbf{x})$  is unknown or too complex for the calculation of the integral. A more practical criterion would be to select the sample that has the largest contribution the current error, that is, the expression under the integral:

$$s = \arg \max_{s \in \mathcal{I}_u} E [(\hat{y}_s - y_s)^2 | \mathbf{x}_s] p(\mathbf{x}_s) \quad (9)$$

If no prior knowledge is available, one can only rely on the error expectation  $E [(\hat{y}_s - y_s)^2 | \mathbf{x}_s]$ . Replacing the probability  $p(y_s | \mathbf{x}_s)$  by its current estimate, one can show that the error expectation is maximal for samples lying on the current decision boundary. More interesting, however, is the case where  $p(\mathbf{x})$  is known and non-uniform. The information about the distribution can then be used to select better data. One possibility to obtain  $p(\mathbf{x})$  is clustering which can be done offline without the interaction with human. The clustering information is useful for active learning due to the two following reasons:

1. The clustering gives extra information for assessing the importance of an unlabeled sample. In particular, the most important are the representative samples located in the center of the clusters, where the density  $p(\mathbf{x})$  is high.
2. Samples in the same cluster are likely to have the same label. This is known as the cluster assumption. Its immediate application is that the class label of one sample can be propagated to the other samples of the same cluster. If so, the active learning can be accelerated as it is sufficient to label just one sample per cluster.

The remainder of the section shows a framework for incorporating pre-clustering information into active learning. In the standard classification, data generation is described by the joint distribution  $p(\mathbf{x}, y)$ . The clustering information can be explicitly incorporated by introducing the hidden cluster indicator  $k \in \{1, \dots, K\}$ , where  $K$  is the number of clusters in the data, and  $k$  indicates that the sample belongs to the  $k$ -th cluster. Assume that all information about the class label  $y$  is already encoded in the cluster indicator  $k$ . This implies that once  $k$  is known,  $y$  and  $\mathbf{x}$  are independent. The joint distribution is written as:

$$p(\mathbf{x}, y, k) = p(y | \mathbf{x}, k) p(\mathbf{x} | k) p(k) = p(y | k) p(\mathbf{x} | k) p(k) \quad (10)$$

$p(y | k)$  is the class probability of a whole cluster. Discriminative probabilistic models can be used to model this probability. In the reported experiment, we have used the logistic regression:

$$p(y|k) = \frac{1}{1 + \exp\{-y(\mathbf{c}_k \cdot \mathbf{a} + b)\}} \quad (11)$$

where  $\mathbf{c}_k$  is the representative of the  $k$ -th cluster which is determined by K-medoid clustering [5]. As such,  $p(y|k)$  specifies a classifier on a representative subset of the data. In the ideal case where the data is well clustered, once all the parameters of  $p(y|k)$  are determined, one could use this probability to determine the label of the cluster representatives, and then assign the same label to the remaining samples in the cluster. In practice, however, such classification will be inaccurate for samples disputed by several clusters. To better classify those samples, a soft cluster membership should be used. This is achieved with the noise distribution  $p(\mathbf{x}|k)$  which propagates the information of label  $y$  from the representatives into the remaining majority of the data. In the experiment,  $p(\mathbf{x}|k)$  are isotropic Gaussians with the same variance for all clusters. As such,  $p(\mathbf{x})$  is mixture of  $K$  Gaussians with the weights  $p(k)$ .

Given the above model, the class posterior is calculated as follows:

$$p(y|\mathbf{x}) = \sum_{k=1}^K p(y, k|\mathbf{x}) = \sum_{k=1}^K p(y|k)p(k|\mathbf{x}) \quad (12)$$

where  $p(k|\mathbf{x}) = p(\mathbf{x}|k)p(k)/p(\mathbf{x})$ . As observed, the classification decision is the weighted combination of the classification results for the representatives with the weights  $p(k|\mathbf{x})$ . Well clustered samples will be assigned the label of the nearest representative. Samples in between the clusters, on the other hand, will be assigned the label of the cluster, which has the highest confidence.

The parameter estimation for the proposed model can also be done by minimizing the same likelihood functions defined in section 2.1. A minor difference is that  $\mathcal{L}^{(1)}$  now depends also on the parameters of the prior data distributions denoted by  $\theta'$ :

$$\mathcal{L}^{(1)} = -\ln p(\theta) - \sum_{i \in \mathcal{I}_\ell} \ln \left\{ \sum_{k=1}^K p(y_i|k; \theta) p(k|\mathbf{x}_i; \theta') \right\} \quad (13)$$

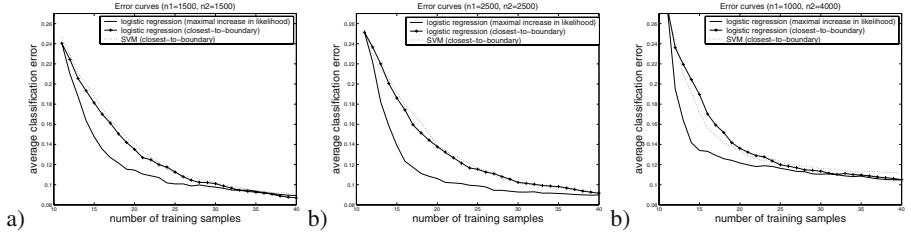
However, the parameters  $\theta'$  are determined mainly by maximizing  $\mathcal{L}^{(1)}$ , since the unlabeled data are overwhelming over the labeled data. The maximization of each likelihood term can therefore be done separately. The clustering algorithm optimizes likelihood  $\mathcal{L}^{(2)}$  to obtain the prior data distribution. The optimization of  $\mathcal{L}^{(1)}$  follows to obtain  $p(y|k)$ .

### 3 Experiment

The proposed algorithms were tested to separate images of human face from non-face images in a database. We have used the database created in [10], which contains 30000 face images of 1000 people and an equal number of non-face images. The images were collected randomly on the Internet and were subsampled to patterns of size 20x20. The face images are carefully aligned to cover the central part of the face. Each original face image is slightly transformed using rotation, scaling and translating, resulting in



**Fig. 1.** Example view of images in the test database.

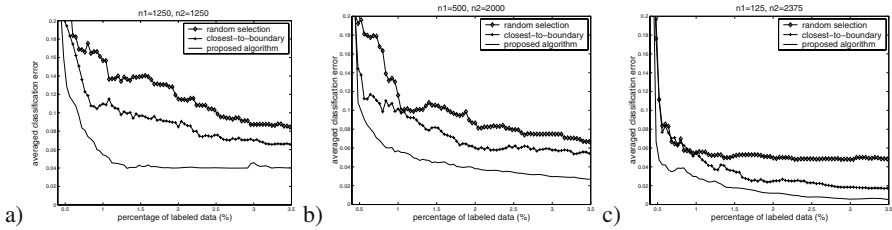


**Fig. 2.** The error curve for the classification of face images using the method proposed in section 2.1. The method is also compared to two other methods that use the closest-to-boundary approach.  $n_1$  and  $n_2$  are the numbers of face and non-face images in the databases respectively.

30 images per person. Example views of some images are shown in Figure 1. From this full database, different test sets were extracted with different settings in number of images and proportions between the numbers of face and non-face images. Each image is regarded as a vector composed of the pixel grey values.

The initial training set contains 5 face and 5 non-face images, and would be added 30-80 more images during the experiment. Each time a new sample is added to the training set, the classifier is re-trained and is tested on the rest of the database. The classification error is calculated as the number of wrong classifications including both missed face samples and false alarms. The performance is evaluated by the decrease of the error as the function of the amount of labeled data. Furthermore, to ensure the accuracy of the results, the error is averaged over number of times of running the algorithms with different randomly selected initial training sets.

Figure 2 shows the performance curves for the method using the maximal increase in likelihood. Figure 3 shows the same curves for the method based on pre-clustering. In both figures, the new methods are compared to two other active learning algorithms which use the closest-to-boundary criterion. It is observed, for instance, in Figure 3a that the best classifier performance levels off to a 4% error. This is the result that would be obtained by labeling all images in the data base. The figure shows that the same result



**Fig. 3.** The results for the classification of face images using pre-clustering as proposed in section 2.2. The proposed method is compared with two SVM-based active learning algorithms that use the closest-to-boundary and random selection.

is achieved with only a small fraction of the database offered for interactive annotation leaving the remainder of the data unlabeled.

## 4 Conclusion

In all results, the new methods show better performance than the existing methods of active learning. That is to say less images need to be offered to the user for labeling to reach the same result. In effect as low as 1% equal to 25 images in the case of Figure 3, need to be labeled (the first 10 selected at random and the remaining 15 selected sequentially on the informativity to the concept boundary) to train the classifier on the concept.

We have reason to believe the improvement tends to be more significant for even larger databases.

## References

1. C. Campbell, N. Cristianini, and A. Smola. Query learning with large margin classifiers. In *Proc. 17th International Conf. on Machine Learning*, pages 111–118. Morgan Kaufmann, CA, 2000.
2. D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Journal of Artificial Intelligence research*, 4:129–145, 1996.
3. J.M. Geusebroek, R. van den Boomgaard, A.W.M. Smeulders, and H. Geerts. Color invariance. *IEEE Trans. on PAMI*, 23(12):1338–1350, December 2001.
4. T. Gevers and A.W.M. Smeulders. Pictoseek: Combining color and shape invariant features for image retrieval. *IEEE Trans. on Image Processing*, 9(1):102, 2000.
5. L. Kaufman and P.J. Rousseeuw. *Finding groups in data: An introduction to cluster analysis*. John Wiley & Sons, 1990.
6. D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In W. Bruce Croft and Cornelis J. van Rijsbergen, editors, *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 3–12. Springer Verlag, 1994.
7. D.G. Lowe. Object recognition from local scale-invariant features. In *Proc. IEEE Conf. on Computer Vision*, pages 1150–1157, 1999.
8. J.L. Mundy and A. Zisserman. *Geometric Invariance in Computer Vision*. MIT Press, 1992.

9. C. Papageorgiou and T. Poggio. A trainable system for object detection. *Int. J. Computer Vision*, 38(1):15–33, 2000.
10. T.V. Pham, M. Worring, and A.W.M. Smeulders. Face detection by aggregated Bayesian network classifiers. *Pattern Recogn. Letters*, 23(4):451–461, 2002.
11. C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *Int. J. Computer Vision*, 37(2):151–172, 2000.
12. A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R.C. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. on PAMI*, 22(12):1349–1380, 2000.
13. K.K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Trans. on PAMI*, 20(1):39–51, 1998.
14. S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *Proceedings of the 9th ACM int. conf. on Multimedia*, pages 107–118, 2001.
15. S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66, Nov. 2001.
16. T. Tuytelaars, A. Turina, and L.J. Van Gool. Noncombinatorial detection of regular repetitions under perspective skew. *IEEE Trans. on PAMI*, 25(4):418–432, 2003.
17. V.N. Vapnik. *Statistical Learning Theory*. Wiley, 1998.
18. P. Viola and M.J. Jones. Robust real-time face detection. *Int. J. Computer Vision*, 57(2):137–154, 2004.
19. M. Worring, A.W.M. Smeulders, and S. Santini. Interaction in content-based retrieval: an evaluation of the state-of-the-art. In R. Laurini, editor, *Advances in Visual Information Systems*, number 1929 in Lect. Notes in Comp. Science, pages 26–36, 2000.
20. T. Zhang and F. J. Oles. Text categorization based on regularized linear classification methods. *Information Retrieval*, 4(1):5–31, 2001.
21. J. Zhu and T. Hastie. Kernel logistic regression and the import vector machine. In *Advances in Neural Information Processing Systems*, 2001.

# An Inference Network Approach to Image Retrieval

Donald Metzler and R. Manmatha

Center for Intelligent Information Retrieval  
Computer Science Department  
University of Massachusetts Amherst  
{metzler, manmatha}@cs.umass.edu,

**Abstract.** Most image retrieval systems only allow a fragment of text or an example image as a query. Most users have more complex information needs that are not easily expressed in either of these forms. This paper proposes a model based on the Inference Network framework from information retrieval that employs a powerful query language that allows structured query operators, term weighting, and the combination of text and images within a query. The model uses non-parametric methods to estimate probabilities within the inference network. Image annotation and retrieval results are reported and compared against other published systems and illustrative structured and weighted query results are given to show the power of the query language. The resulting system both performs well and is robust compared to existing approaches.

## 1 Introduction

Many existing image retrieval systems retrieve images based on a query image [1]. However, recently a number of methods have been developed that allow images to be retrieved given a text query [2,3,4]. Such methods require a collection of training images annotated with a set of words describing the image's content. From a user's standpoint, it is generally easier and more intuitive to produce a text query rather than a query image for a certain information need. Therefore, retrieval methods based on textual queries are desirable.

Unfortunately, most retrieval methods that allow textual queries use a rudimentary query language where queries are posed in natural language and terms are implicitly combined with a soft Boolean AND. For example, in current models, the query **tiger jet** allows no interpretation other than **tiger AND jet**. What if the user really wants **tiger OR jet**? Such a query is not possible with most existing models. A more richly structured query language would allow such queries to be evaluated.

Finally, an image retrieval system should also allow seamless combination of text and image representations within queries. That is, a user should be able to pose a query that is purely textual, purely based on an image, or some combination of text and image representations.

This paper presents a robust image retrieval model based on the popular Inference Network retrieval framework [5] from information retrieval that successfully combines all of these features. The model allows structured, weighted queries made up of both textual and image representations to be evaluated in a formal, efficient manner.

We first give a brief overview of related work, including a discussion of other image retrieval methods and an overview of the Inference Network retrieval framework in Section 2. Section 3 details our model. We then describe experimental results and show retrieval results from several example queries in Section 4. Finally, we discuss conclusions and future work in Section 5.

## 2 Related Work

The model proposed in this paper draws heavily from past work in the information and image retrieval fields. This section gives an overview of related work from both fields. A number of models use associations between terms and image regions for image annotation and retrieval. The *Co-occurrence Model* [6] looks at the co-occurrences of annotation words and rectangular image regions to perform image annotation. The *Translation Model* [7] uses a classic machine translation technique to translate from a vocabulary of terms to a vocabulary of blobs. Here, blobs are clusters of feature vectors that can be thought of as representing different “concepts”. An unannotated image is represented as a set of blobs and translated into a set of annotation words. The *Cross-Media Relevance Model* [3] (CMRM) also views the task as translation, but borrows ideas from cross-lingual information retrieval [8], and thus allows for both image annotation and retrieval. The *Correspondence LDA Model* [2] (CLDA) allows annotation and retrieval. It is based on Latent Dirichlet Allocation [9] and is a generative model that assumes a low dimensional set of latent factors generate, via a mixture model, both image regions and annotations.

The motivation for the estimation techniques presented here is the *Continuous Relevance Model* [4] (CRMs), which is a continuous version of the CMRM model that performs favorably. Unlike other models that impose a structure on the underlying feature space via the use of blobs, the CRM model uses a non-parametric technique to estimate the joint probability of a query and an image. However, it assumes annotation terms are drawn from a multinomial distribution, which may be a poor assumption. Our estimation technique makes no assumption as to the distribution of annotation terms and thus is fully non-parametric.

The Inference Network retrieval framework is a robust model from the field of information retrieval [5] based on the formalism of Bayesian networks [10]. Some strong points of the model are that it allows structured, weighted queries to be evaluated, multiple document representations, and efficient inference. One particular instantiation of the inference network framework is the InQuery retrieval system [11] that once powered Infoseek and currently powers the THOMAS search engine used by the Library of Congress. Additionally, inference networks

for multimedia retrieval in extensible databases have been explored [12]. Experiments have shown that intelligently constructed structured queries can translate into improved retrieval performance. Therefore, the Inference Network framework is a good fit for image retrieval.

### 3 Image Inference Network Model

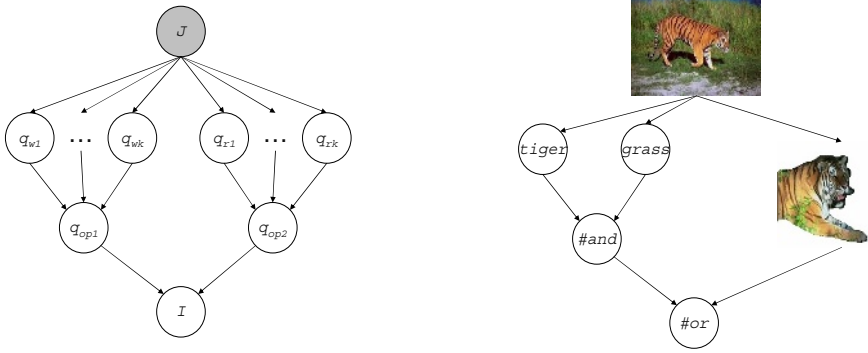
Suppose we are given a collection of annotated training images  $\mathcal{T}$ . Each image  $I \in \mathcal{T}$  has a fixed set of words associated with it (annotation) that describe the image contents. These words are encoded in a vector  $tf_I$  that contains the number of times each word occurs in  $I$ 's annotation. We also assume that  $I$  has been automatically segmented into regions. Note that each image may be segmented into a different number of regions. A fixed set of  $d$  features is extracted from each of these regions. Therefore, a  $d$  dimensional feature vector  $r_i$  is associated with each region of  $I$ . Finally, each feature vector  $r_i$  extracted from  $I$  is assumed to be associated with each word in  $I$ 's annotation. Therefore,  $tf_I$  is assumed to describe the contents of each  $r_i$  in  $I$ . Images in the test set are represented similarly, except they lack annotations. Given a set of unseen test images, the image retrieval task is to return a list of images ranked by how well each matches a user's information need (query).

#### 3.1 Model

Our underlying retrieval model is based on the Inference Network framework [5]. Figure 1 (left) shows the layout of the network under consideration. The node  $J$  is continuous-valued and represents the event an image is described by a collection of feature vectors. The  $q_{w_j}$  nodes are binary and represent the event that word  $w_j$  is observed. Next, the  $q_{r_k}$  nodes are binary and correspond to the event that image representation (feature vector)  $r_k$  is observed. Finally, the  $q_{op}$  and  $I$  nodes represent query operator nodes.  $I$  is simply a special query operator that combines all pertinent evidence from the network into a single belief value representing the user's information need. These nodes combine evidence about word and image representation nodes in a structured manner and allow efficient marginalization over their parents [13]. Therefore, to perform ranked image retrieval, for each image  $X$  in the test collection we set  $J = X$  as our observation, run belief propagation, and rank documents via  $P(I|X)$ , the probability the user's information need is satisfied given that we observe image  $X$ .

Figure 1 (right) illustrates an example instantiation of the network for the query `#OR(#AND(tiger grass) <tiger.jpg>)`. This query seeks an image with both `tigers` AND `grass` OR an image that is similar to `tiger.jpg`. The image of the tiger appearing at the top is the image currently being scored. The other nodes, including the cropped tiger image, are dynamically created based on the structure and content of the query. Given estimates for all the conditional probabilities within the network, inference is done, and the document is scored based on the belief associated with the `#OR` node.





**Fig. 1.** Inference network layout (left) and example network instantiation (right)

Now that the inference network has been set up, we must specify how to compute  $P(q_w|J)$ ,  $P(q_r|J)$ , and the probabilities at the query operator nodes. Although we present specific methods for estimating the probabilities within the network, the underlying model is robust and allows any other imaginable estimation techniques to be used.

### 3.2 Image Representation Node Estimation

To estimate the probability of observing an image representation  $r$  given an image  $J$  we use a density estimation technique. The resulting estimate gives higher probabilities to representations that are “near” (or similar to), on average, the feature vectors that represent image  $J$ . Thus, the following estimate is used:

$$P(q_r|J) = \frac{1}{|r_J|} \sum_{r_i \in J} \mathcal{N}(q_r; r_i, \Sigma)$$

where  $|r_J|$  is the number of feature vectors associated with  $J$  and,

$$\mathcal{N}(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp \left[ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right]$$

is a multivariate Gaussian kernel. Here,  $\Sigma$  is assumed to be diagonal and is the empirical variance with respect to the training set feature vectors. Note that any appropriate kernel function can be used in place of  $\mathcal{N}$ .

### 3.3 Term Node Estimation

Estimating the likelihood a term is observed given an image,  $P(q_w|J)$ , is a more difficult task since test images are not labeled with annotations. Inverting the

probability by Bayes' rule and assuming feature vectors are conditionally independent given  $q_w$  we see that

$$P(q_w|J) \propto P(q_w)P(J|q_w) = P(q_w) \prod_{r_i \in J} P(r_i|q_w)$$

where  $P(q_w) = \frac{n_{q_w}}{n_{tot}}$ ,  $n_{q_w}$  is the number of feature vectors in the training set  $q_w$  is associated with and  $n_{tot}$  is the total number of feature vectors in the training set. To compute  $P(r_i|q_w)$  we again use density estimation to estimate the density of feature vectors among the training set that are associated with term  $q_w$ . Then,

$$P(r_i|q_w) = \frac{1}{n_{q_w}} \sum_{\substack{I \in \mathcal{T} \\ tf_I(q_w) > 0}} \sum_{g_k \in I} \mathcal{N}(r_i; g_k, \Sigma)$$

where  $tf_I(q_w)$  indicates the number of times that  $q_w$  appears in image  $I$ 's annotation and  $\mathcal{N}$  is defined as above. Finally, it should be noted that when  $\Sigma$  is estimated from the data our model does not require hand tuning any parameters.

### 3.4 Regularized Term Node Estimates

As we will show in Section 4, the term node probability estimates used here result in good annotation performance. This indicates that our model estimates probabilities well for single terms. However, when combining probabilities from multiple term nodes we hypothesize that it is often the case that the *ordering* of the term likelihoods captures more important information than their actual values. That is, the fact that  $tiger = \arg \max_{q_w} P(q_w|J_1)$  is more important than the fact that  $P(tiger|J_1) = 0.99$ . Thus, we explore regularizing the term node probabilities for each image so they vary more uniformly and are based on the ordering of the term likelihoods.

Assuming the term likelihood ordering is important and that for an image: 1) a few terms are very relevant (correctly annotation terms), 2) a medium number of terms are somewhat relevant (terms closely related to the annotation terms), and 3) a large number of terms are irrelevant (all the rest). Following these assumptions we fit the term probability estimates to a Zipfian distribution. The result is a distribution where a large probability mass is given to relevant terms, and a smaller mass to the less relevant terms. We assume that terms are relevant based on their *likelihood rank*, which is defined as the rank of term  $w$  in a sorted (descending) list of terms according to  $P(q_w|J)$ . Therefore, the most likely term is given rank 1. For an image  $J$  we regularize the corresponding term probabilities according to  $\hat{P}(q_w|J) = Z^{-1} \frac{1}{R_{q_w,J}}$  where  $R_{q_w,J}$  is the likelihood rank of term  $w$  for image  $J$  and  $Z^{-1}$  normalizes the distribution.

### 3.5 Query Operators

Query operators allow us to efficiently combine beliefs about query word nodes and image representation nodes in a structure (logical) and/or weighted manner. They are defined in such a way as to allow efficient marginalization over

their parent nodes [14], which results in fast inference within the network. The following list represents a subset of the structured query operators available in the InQuery system that have been implemented in our system: **#AND**, **#WAND**, **#OR**, **#NOT**, **#SUM**, and **#WSUM**. To compute  $P(q_{op} = true|J)$ , the belief at query node  $q_{op}$ , we use the following:

$$\begin{aligned} P_{\#AND}(q_{op}|J) &= \prod_i p_i & P_{\#WAND}(q_{op}|J) &= \prod_i p_i^{\frac{w_i}{W}} \\ P_{\#SUM}(q_{op}|J) &= \frac{1}{n} \sum_i p_i & P_{\#WSUM}(q_{op}|J) &= \frac{1}{W} \sum_i w_i p_i \\ P_{\#OR}(q_{op}|J) &= 1 - \prod_i (1 - p_i) & P_{\#NOT}(q_{op}|J) &= 1 - p_1 \end{aligned}$$

where node  $q_{op}$  has  $n$  parents  $\pi_1, \dots, \pi_n$ ,  $p_i = P(\pi_i|J)$ , and  $W = \sum_i w_i$ . See [11, 14, 13] for a derivation of these expressions, an explanation of these operators, and details on other possible query operators.

## 4 Results

We tested our system using the Corel data set that consists of 5000 annotated images. Each image is annotated with 1-5 words. The number of distinct words appearing in annotations is 374. The set is split into 4500 training images and 500 test images. Each image is segmented using normalized cuts into 1-10 regions. A standard set of 36 features based on color and texture is extracted from each image region. See [7] for more details on the features used. We compare the results of our system with those reported from the Translation, CMRM and CRM models that used the same segmentation and image features. Throughout the results, **InfNet-reg** refers to the model with regularized versions of the term probabilities and **InfNet** refers to it with unregularized probabilities.

### 4.1 Image Annotation

The first task we evaluate is image annotation. Image annotation results allow us to compare how well different methods estimate term probabilities. The goal is to annotate unseen test images with the 5 words that best describe the image. Our system annotates these words based on the 5 terms with the highest likelihood rank for each image. Mean per-word recall and precision are calculated, where recall is the number of images correctly annotated with a word divided by the number of images that contain that word in the human annotation, and precision is the number of images correctly annotated with a word divided by the total number of images annotated with that word in the test set. These metrics are computed for every word and the mean over all words and are reported in Table 1. As the table shows, our system achieves very good performance on the annotation task. It outperforms CRMs both in terms of mean word precision and recall, with the mean per-word recall showing a 26.3% improvement over the CRM model.

**Table 1.** Annotation results

Models	Translation	CMRM	CRM	InfNet
# words w/ recall > 0	49	66	107	112
Results on full vocabulary				
Mean per-word recall	0.04	0.09	0.19	0.24
Mean per-word precision	0.06	0.10	0.16	0.17

## 4.2 Image Retrieval

For the retrieval task we create all 1-, 2-, 3- word queries that are relevant to at least 2 test images. An image is assumed to be relevant if and only if its annotation contains every word in the query. Then, for a query  $Q = q_1, q_2, \dots, q_L$  we form and evaluate a query of the form  $\#and(q_1, \dots, q_L)$ . We use the standard information retrieval metrics of mean average precision (MAP) and precision at 5 ranked documents to evaluate our system. Table 2 reports the results.

**Table 2.** Retrieval results and comparison

Query length	1 word	2 word	3 word
Number of queries	179	386	178
Relevant images	1675	1647	542
Precision after 5 retrieved images			
CMRM	0.1989	0.1306	0.1494
CRM	0.2480	0.1902	0.1888
InfNet	0.2525	0.1672	0.1727
InfNet-reg	0.2547	0.1964	0.2170
Mean Average Precision			
CMRM	0.1697	0.1642	0.2030
CRM	0.2353	0.2534	0.3152
InfNet	0.2484	0.2155	0.2478
InfNet-reg	0.2633	0.2649	0.3238

The regularized probabilities result in much better retrieval performance over the unregularized probabilities. Using these probabilities, our system achieves better performance than both the CMRM and CRM models on all 3 query sets for both the MAP and precision after 5 retrieved documents metric.

Figure 2 gives illustrative examples of the top 4 ranked documents using the regularized estimates for several structured queries. The first query of Figure 2,  $\#or(swimmers\ jet)$ , results in images of both swimmers and jets being returned. The next query shows that a standard query gives good retrieval results. The next two queries demonstrate how term weighting can affect the retrieval results. Finally, the last query shows an example query that mixes text and image representations, with the bird image being part of the query. These results show that the structured operators can be used to form rich, powerful queries that other approaches are not capable of.

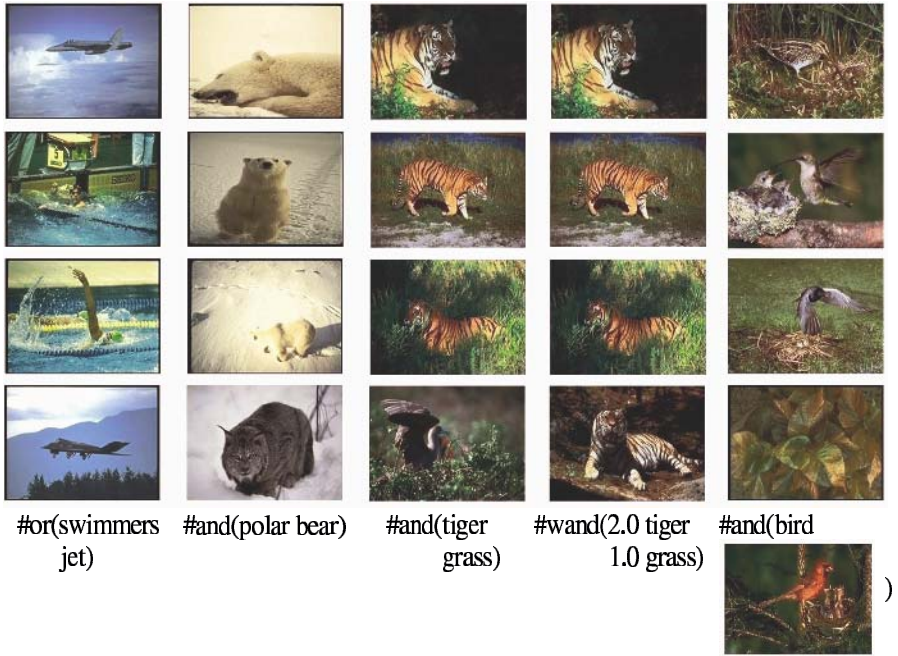


Fig. 2. Example structured query results

## 5 Conclusions and Future Work

We have presented an image retrieval system based on the inference network framework from information retrieval. The resulting model allows rich queries with structured operators and term weights to be evaluated for combinations of terms and images. We also presented novel non-parametric methods for estimating the probability of a term given an image and a method for regularizing the probabilities. Our system performs well compared to other published models under standard experimental evaluation.

There are a number of things that can be done as part of future work. First, better estimates for  $P(q_r|J)$  are needed. The method described in this paper was used for simplicity. Second, our system must be tested using different segmentation and better features to allow comparison against other published results. Third, more rigorous experiments should be done using the structured and weighted query operators to show empirically what affect they have on overall performance. Finally, it would be interesting to explore a model that combines the current system with a document retrieval system to allow for full text and image search in a combined model.

**Acknowledgments.** We thank Kobus Barnard for making the Corel dataset available at [http://vision.cs.arizona.edu/kobus/research/data/eccv\\_2002/](http://vision.cs.arizona.edu/kobus/research/data/eccv_2002/).

This work was supported in part by the Center for Intelligent Information Retrieval and in part by the National Science Foundation under grant number IIS-9909073 and in part by SPAWARSYSCEN-SD grant number N66001-02-1-8903. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

## References

1. Flickner, M., Sawhney, H.S., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., Yanker, P.: Query by image and video content: The QBIC system. *IEEE Computer Magazine* **28** (Sept. 1995) 23–30
2. Blei, D.M., Jordan, M.I.: Modeling annotated data. In: *Proceedings of the 26th annual international ACM SIGIR conference*. (2003) 127–134
3. Jeon, J., Lavrenko, V., Manmatha, R.: Automatic image annotation and retrieval using cross-media relevance models. In: *Proceedings of the 26th annual international ACM SIGIR conference*. (2003) 119–126
4. Lavrenko, V., Manmatha, R., Jeon, J.: A model for learning the semantics of pictures. In: *Proceedings of the Seventeenth Annual Conference on Neural Information Processing Systems*. (2003)
5. Turtle, H., Croft, W.B.: Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems (TOIS)* **9** (1991) 187–222
6. Mori, Y., Takahashi, H., Oka, R.: Image-to-word transformation based on dividing and vector quantizing images with words. In: *MISRM'99 First International Workshop on Multimedia Intelligent Storage and Retrieval Management*. (1999)
7. Duygulu, P., Barnard, K., de Fretias, N., Forsyth, D.: Object recognition as machine translation: Learning a leicon for a fixed image vocabulary. In: *Seventh European Conference on Computer Vision*. (2002) 97–112
8. Lavrenko, V., Choquette, M., Croft, W.: Cross-lingual relevance models. In: *Proceedings of the 25th annual International ACM-SIGIR conference*. (2002)
9. Blei, D., Ng, A., Jordan, M.: Latent dirichlet allocation. *Journal of Machine Learning Research* **3** (2003) 993–1022
10. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann Publishers Inc. (1988)
11. Callan, J.P., Croft, W.B., Harding, S.M.: The INQUERY retrieval system. In: *Proceedings of DEXA-92, 3rd International Conference on Database and Expert Systems Applications*. (1992) 78–83
12. de Vries, A.: Mirror: Multimedia query processing in extensible databases. In: *Proceedings of the fourteenth Twente workshop on language technology (TWLT14)*. (1998) 37–48
13. Turtle, H.R.: Inference Networks for Document Retrieval. PhD thesis, University of Massachusetts (1991)
14. Greiff, W.R., Croft, W.B., Turtle, H.: PIC matrices: a computationally tractable class of probabilistic query operators. *ACM Transactions on Information Systems* **17** (1999) 367–405

# Small Sample Size Performance of Evolutionary Algorithms for Adaptive Image Retrieval

Zoran Stejić<sup>1</sup>, Yasufumi Takama<sup>2</sup>, and Kaoru Hirota<sup>1</sup>

<sup>1</sup> Dept. of Computational Intelligence and Systems Science  
Interdisciplinary Graduate School of Science and Engineering  
Tokyo Institute of Technology

G3-49, 4259 Nagatsuta, Midori-ward, Yokohama 226-8502, Japan  
stejic@hrt.dis.titech.ac.jp

<sup>2</sup> Dept. of Electronic Systems Engineering  
Tokyo Metropolitan Institute of Technology, Tokyo, Japan

**Abstract.** We evaluate the small sample size (SSS) performance of evolutionary algorithms (EAs) for relevance feedback (RF) in image retrieval. We focus on the requirement to learn the user's information need based on a small — between 2 and 25 — number of positive and negative training images. Despite this being a fundamental requirement, none of the existing works dealing with EAs for RF systematically evaluates their SSS performance. To address this issue, we compare four variants of EAs for RF. Common for all variants is the hierarchical, region-based image similarity model, with region and feature weights as parameters. The difference between the variants is in the objective function of the EA used to adjust the model parameters. The objective functions include: (O-1) precision; (O-2) average rank; (O-3) ratio of within-class (i.e., positive images) and between-class (i.e., positive and negative images) scatter; and (O-4) combination of O-2 and O-3. We note that — unlike O-1 and O-2 — O-3 and O-4 are not used in any of the existing works dealing with EAs for RF. The four variants are evaluated on five test databases, containing 61,895 general-purpose images, in 619 semantic categories. Results of the evaluation reveal that variants with objective functions O-3 and O-4 consistently outperform those with O-1 and O-2. Furthermore, comparison with the representative of the existing RF methods shows that EAs are both effective and efficient approaches for SSS learning in region-based image retrieval.

## 1 Introduction

Modeling image similarity is one of the most important issues in the present image retrieval research [12,15]. When asked to retrieve the database images similar to the user's query image(s), the retrieval system must *approximate the user's similarity criteria*, in order to identify the images which satisfy the user's information need.

Given that the user's similarity criteria are both subjective and context-dependent [12,17], the adaptation of the retrieval system to the user — through

the relevance feedback (RF) — is crucial [17]. The objective of RF is to improve the retrieval performance by learning the user’s information need based on the interaction with the user. RF is “shown to provide dramatic performance boost in [image] retrieval systems,” and even “it appears to have attracted more attention in the new field [image retrieval] than in the previous [text retrieval].” [17]

Among a variety of machine learning and pattern recognition techniques applied to RF in information retrieval in general [17,6], probably the greatest imbalance between text and image retrieval is in the applications of evolutionary algorithms (EAs). Namely, while EAs are widely used and proven to be both effective and efficient approaches to RF in text retrieval [3,6], their applications to RF in image retrieval are still rare [17].

Our **objective** in this paper is to evaluate the performance of EAs for RF in image retrieval. We compare four variants of EAs for RF — two of the existing ones, and the two new ones we introduce. We particularly focus on the small sample size (SSS) performance, i.e., the requirement for a retrieval system to learn the user’s information need based on a small — up to a few tens — number of positive (i.e., relevant for the user’s information need) and negative (i.e., irrelevant) training images. Despite this being a fundamental requirement in the RF context [17], none of the existing works dealing with EAs for RF systematically evaluates their SSS performance.

The four variants of EAs for RF are evaluated on five test databases containing 61,895 general-purpose images, in 619 semantic categories. In total, over 1,000,000 queries are executed, based on which the (weighted) precision, average rank, and retrieval time are computed. The comparison with the representative of the existing RF methods is performed as well.

The main **contributions** of the present work are: (1) *from the viewpoint of EAs for RF in image retrieval*: (a) we perform the first systematic evaluation of their SSS performance; (b) we introduce two new EA variants, shown to outperform the existing ones; (2) *from the viewpoint of RF in image retrieval in general*: through the comparison with the existing RF methods, we demonstrate that EAs are both effective and efficient approaches for SSS learning in region-based image retrieval.

The rest of the paper is structured as follows. In Section 2, the related works are surveyed, dealing with the RF and EAs in image retrieval. Next, in Section 3, the four variants of EAs for RF are introduced. Finally, in Section 4, the evaluation is described, of the SSS performance of the four variants.

## 2 Related Works: RF and EAs in Image Retrieval

**Evolutionary algorithms.** EAs refer to a class of *stochastic, population-based search algorithms* [9,5], including: *genetic algorithms (GAs)*, *evolution strategies (ESs)*, *evolutionary programming (EP)*, etc.

Regarding their desirable characteristics — from the viewpoint of application to the learning and/or optimization tasks — EAs are [9,5]: (1) global search algorithms, not using the gradient information about the objective function, which



makes them applicable to nondifferentiable, discontinuous, nonlinear, and multi-modal functions; (2) robust in that their performance does not critically depend on the control parameters; (3) easy to implement, since the basic mechanism of an EA consists of a small number of simple steps; (4) domain-independent, i.e., “weak,” in that they do not use the domain knowledge; and (5) flexible in that they can equally handle both continuous and discrete variables to be optimized.

Regarding the less desirable characteristics, EAs are in general: (1) not guaranteed to find a globally optimal solution to a problem; and (2) computationally intensive. However, in practice [9], the sub-optimal solutions found by EAs are shown to be “sufficiently good” in most cases, while the parallel implementation — natural for EAs — can solve the computation time problem.

**EAs in text and image retrieval.** Among a variety of techniques applied to RF in information retrieval in general, probably the greatest imbalance between text and image retrieval is in the applications of EAs.

On one hand, applications of *EAs to RF in text retrieval* are numerous — they are even covered in a well-known and widely used information retrieval textbook [6], and are also a subject of a recent comprehensive survey containing 57 references [3]. On the other hand, works dealing with *EAs for RF in image retrieval* are still rare [2,13] — none is mentioned among the 64 references in a recent comprehensive survey of RF in image retrieval [17].

The issue not sufficiently addressed in the works dealing with EAs for RF in image retrieval is the SSS, despite this being a fundamental requirement in the RF context. As the objective functions — computed over the training images — either the retrieval precision [13], or the average rank of the positive images [2] are used, neither of which is suitable for the SSS learning. Furthermore, no work systematically evaluates the SSS performance of EAs for RF in image retrieval.

**Distinguishing characteristics of EAs for RF.** In the context of RF in image retrieval, the distinguishing characteristics of EAs are: (1) unlike majority of other RF techniques performing the “transformation of the feature space” [17], EAs perform the “modification of the similarity model”; (2) arbitrary nonlinearities or complex mathematical operators can be easily incorporated in the similarity model, without changing the EA used to adjust the model parameters; and (3) the learning algorithm is inherently incremental, suitable for long-term learning (i.e., creating user profiles) as well.

Furthermore, the computation time of EAs is not a problem, since the number of training samples is small — as we experimentally demonstrate in Section 4.

### 3 Four Variants of Evolutionary Algorithms for Relevance Feedback

We introduce four variants of EAs for RF in image retrieval. Common for all variants is the hierarchical, region-based image similarity model, with region and feature weights as parameters (Section 3.1). The difference between the variants

is in the EA used to adjust the model parameters (Section 3.2), based on the positive and negative training images.

### 3.1 Hierarchical Region-Based Image Similarity Model

In the proposed image similarity model, image similarity is expressed as a weighted arithmetic mean of region similarities, while each region similarity is expressed as a weighted arithmetic mean of the corresponding feature similarities. Variables and functions used for the formalization of the proposed model are explained in the following.

**Set of images.**  $I$  represents the image database. Area of each image is uniformly partitioned into  $N \times N (= n_R)$  regions — i.e., rectangular blocks of equal size — defined by the **set of regions**  $R$ . From each region,  $n_F$  features are extracted, defined by the **set of features**  $F$ . Based on the extracted features — e.g., color or texture — similarity of a pair of image regions is computed. A collection of (feature or region) similarity values are mapped into a single, overall (region or image) similarity value, using the weighted arithmetic mean.

The hierarchical image similarity computation is expressed by the following equation ( $\mathbf{q}$  is the query,  $\mathbf{i}$  is the database image —  $\mathbf{q}, \mathbf{i} \in I$ ):

$$\underbrace{S_I(\mathbf{q}, \mathbf{i})}_{\text{image similarity}} = \sum_{r \in R} \underbrace{w_R(\mathbf{q}, \mathbf{i}; r)}_{\text{region weight}} \overbrace{\sum_{f \in F} \underbrace{w_F(\mathbf{q}, \mathbf{i}; r; f)}_{\text{feature weight}} \underbrace{S_F(\mathbf{q}, \mathbf{i}; r; f)}_{\text{feature similarity}}}^{\text{region similarity}} \quad (1)$$

Regarding the number of regions (set  $R$ ) into which each image is partitioned, in the experiments we used  $N = 4$ , i.e.,  $n_R = N \times N = 16$  regions (Section 4). While the choice is heuristic, it results in a satisfactory retrieval performance, and other region-based image similarity models use a similar number of regions [15, 13].

Regarding the image features (set  $F$ ) extracted from each region, based on which feature similarity values  $S_F(\mathbf{q}, \mathbf{i}; \mathbf{r}; \mathbf{f})$  are computed, we have chosen *three features most commonly used in the image retrieval* [8]: color, shape, and texture. Three features imply that  $n_F = 3$ . Color features are represented by *color moments* [14], resulting in a 9-D feature vector. Shape features are represented by *edge-direction histogram* [1], resulting in a 8-D feature vector. Texture features are represented by *texture neighborhood* [7], resulting in a 8-D feature vector. In total, each image region is represented by 9-D+8-D+8-D = 25-D feature vector. The feature similarity values  $S_F(\mathbf{q}, \mathbf{i}; \mathbf{r}; \mathbf{f})$  are inversely proportional to the *distance between the corresponding feature vectors*, which are computed using the (weighted) *city-block distance* [8] for all three features.

### 3.2 Four Variants of EA for Adjusting Model Parameters

The four EA variants are used to adjust the region and feature weights of the proposed similarity model, based on the positive and negative training images

provided by the user through the interaction. The difference between the four variants is in the objective function of the EA. The EA itself is the same for all variants — the *evolution strategy (ES)*, with a modification we introduce.

**Evolutionary algorithm.** We choose the ES among EAs for two reasons: (1) ESs are particularly suitable for the real parameter (in our case, weights) optimization [9]; and (2) ESs are the simplest and the easiest to implement among the EAs.

We employ a *two-membered evolution strategy* — (1+1)ES — which is the basic ES. The underlying idea is to: (1) randomly generate an initial solution; and (2) iteratively generate new solutions by applying the stochastic mutation operator to the current solution. Whenever a new solution is generated, it competes with the current one, and replaces it in the case it is better — as evaluated by the objective function. The process continues for a predefined number of iterations, or until a sufficiently good solution is found.

We modify the original ES by combining two types of mutation — *uniform random mutation* and *Gaussian random mutation*. In general, the uniform mutation is more suitable for global search, while the Gaussian mutation is more suitable for local search [9]. To additionally emphasize these characteristics we: (1) set the mutation rate of the uniform mutation to a high value, and discretize the weight space on which the uniform mutation operates; and (2) set the mutation rate of the Gaussian mutation to a low value, while using the original, continuous weight space. Whenever the uniform mutation — performing global search — generates a new solution which is better than the current one, the Gaussian mutation is iteratively applied to the new solution, to perform local search, i.e., “fine tuning.” In this way, the global and local search are more effectively combined than in the case of the original ES.

**Objective functions.** As the objective functions — computed over the training images — we use: (O-1) precision; (O-2) average rank; (O-3) ratio of positive and negative scatter; and (O-4) combination of O-2 and O-3.

Within each objective function, the training images are ranked based on the similarity to the centroid of positive images, using the image similarity function (Equation 1). Given  $n_P$  positive and  $n_N$  negative training images, precision is computed as the fraction of positive images, among the top-ranked  $n_P$  images. Average rank is simply the average of the rank values for positive images. The positive and negative scatter are defined as the average similarity (computed using Equation 1) between the centroid of positive images, and: (1) each of the positive images, (2) each of the negative images — respectively.

Functions O-1 [13] and O-2 [2] are used in the existing works dealing with EAs for RF. It follows from the definition of the two functions that O-2 is more sensitive to a change in ranking among the training images — thus being more suitable for the SSS learning than O-1. On the other hand, the newly introduced function O-3 is inspired by the Biased Discriminant Analysis (BDA) [16], and is expected to be more suitable for the SSS learning than O-2.

## 4 Experimental Evaluation of Small Sample Size Retrieval Performance

### 4.1 Experimental Setting

**Test databases.** Five test databases are used, containing 61,895 images, in 619 semantic categories: (1) **Corel-1000-A database** [15], with 1,000 images in 10 categories; (2) **Corel-1000-B database** [13], with 1,000 images in 10 categories; (3) **Corel-20k-A database**, with 20,000 images in 200 categories; (4) **Corel-20k-B database**, with 19,998 images in 200 categories; and (5) **Corel-20k-C database**, with 19,897 images in 199 categories. The number of images per category varies between 97 and 100 for the five databases. Corel-20k-A, Corel-20k-B, and Corel-20k-C databases are obtained by partitioning the Corel-60k database [15] into three approximately equal subsets. All the five databases are subsets of the Corel image collection [4], and contain color photographs, ranging from natural scenes to artificial objects. The Corel collection is well-known and frequently used for the evaluation of the image retrieval methods [15,13].

Partitioning of each database into semantic categories is determined by the creators of the database, and reflects the human perception of image similarity. The semantic categories define the **ground truth** in a way that, for a given query image, the relevant images are considered to be those — and only those — that belong to the same category as the query image.

**Performance measures.** The performance measures we use are: (1) precision, (2) weighted precision, (3) average rank, and (4) retrieval time. These are the four most frequently used measures of the image retrieval performance [12]. All the four performance measures are computed for each query, based on the given ground truth. The performance measures are averaged for each image category, as well as for each test database.

**Queries.** A **query** consists of a set of positive and negative training images. In the **basic test cases**, the number of positive (P) and negative (N) training images is varied between  $1P+1N$  and  $13P+12N$ , in 24 steps:  $1P+1N$ ,  $2P+1N$ ,  $2P+2N$ ,  $3P+2N$ ,  $\dots$ ,  $13P+12N$ . Accordingly, there are 24 basic test cases.

Each basic test case is executed 20 times for each image category, in each test database. Each time, a different set of positive and negative training images is randomly chosen, with positive images being from the given category, and negative from other categories. This gives  $(\#categories \times \#basic\ test\ cases \times 20\ trials)$  queries per test database, i.e.,  $619 \times 24 \times 20 = 297,120$  queries in total.

In the **additional test cases**, the number of positive (i.e., negative) training images is fixed to 1, 2, and 3, respectively, while the number of negative (i.e., positive) training images is varied between 1 and 10, i.e.:  $1P+1N$ ,  $1P+2N$ ,  $\dots$ ,  $1P+10N$ ,  $2P+1N$ ,  $2P+2N$ ,  $\dots$ ,  $2P+10N$ ,  $3P+1N$ ,  $3P+2N$ ,  $\dots$ ,  $3P+10N$ . This gives 60 additional test cases — 30 with small number of positive and increasing number of negative images, and 30 for the opposite situation. As for the basic test cases, each additional test case is executed 20 times for each image category in each test database, resulting in  $619 \times 60 \times 20 = 742,800$  queries in total.

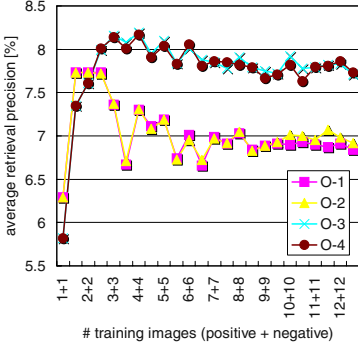
**Comparison with existing RF methods.** Besides comparing the retrieval performance of the four EA variants with objective functions O-1–O-4, we also compare the best-performing of the four variants with the two of the existing RF methods — Query Point Movement (QPM) method [10] and Standard Deviation-Based (StDev) method [11]. The two methods are chosen since they are easy to implement, and are shown to be both efficient and effective [10,11]. The image features used in the two methods are the same as those used in the proposed method (Section 3.1).

## 4.2 Experiment Results and Discussion

Experiment results are summarized in Figure 1.

(a) Overall comparison of EA variants

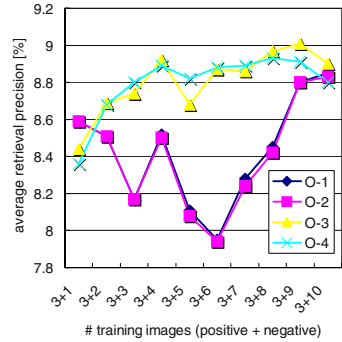
<i>Perform.</i>	<i>Objective Function</i>			
<i>Measure</i>	O-1	O-2	O-3	O-4
P. [%]	23.86	23.96	24.79	24.98
W.P. [%]	30.45	30.60	31.96	32.15
A.R.	3373	3346	3318	3298



(c) Evaluation on Corel-20k-B DB

(b) Best variant vs. existing methods

<i>Test</i>	<i>Method</i>		
<i>Database</i>	QPM	StDev	O-4
Corel-1000-A	49.01	52.72	55.50
Corel-1000-B	39.29	42.72	43.51
Corel-20k-A	5.31	6.28	8.43
Corel-20k-B	4.62	5.60	7.75
Corel-20k-C	6.13	7.44	9.71



(d) Evaluation on Corel-20k-A DB

**Fig. 1.** (a) Comparison of the four EA variants, with objective functions O-1–O-4. For each variant, average values of retrieval precision (P.), weighted precision (W.P.), and average rank (A.R.) are given. Each value is an average of 297,120 queries, corresponding to the 24 basic test cases evaluated over the five test databases. (b) Comparison of the best-performing of the four EA variants (O-4), with the QPM and StDev methods. For each method, the average retrieval precision is given, evaluated on each of the five test databases. (c) Comparison of the four EA variants on Corel-20k-B database. For each variant, average retrieval precision for the 24 basic test cases is given. Each value is an average of 4,000 queries (200 categories  $\times$  20 trials per category). (d) Comparison of the four EA variants on Corel-20k-A database. For each variant, average retrieval precision for the 10 of the additional test cases is given. Each value is an average of 4,000 queries (200 categories  $\times$  20 trials per category).

The average retrieval time per image for the four EA variants is in the range [0.1sec, 0.8sec], depending on the number of training images, evaluated on a 750MHz Pentium III processor.

Based on the experiment results, the following observations can be made:

1. EA variants with objective functions O-3 and O-4 consistently outperform the variants with objective functions O-1 and O-2 (Figure 1-a, c, d). This demonstrates that the newly introduced objective functions (O-3 and O-4) are more suitable for the SSS learning than the existing ones (O-1 and O-2).
2. The objective function O-4 — combining objective functions O-2 and O-3 — results in the best performing EA variant. In general, EAs allow for straightforward combination of the objective functions.
3. The best performing EA variant (O-4) consistently outperforms the representative of the existing relevance feedback methods, on all test databases (Figure 1-b). This is a consequence of both: (1) the proposed image similarity model, with the region and feature weights as parameters — as opposed to QPM and StDev methods with only the feature weights as parameters; and (2) the EA used to adjust the model parameters.
4. Increasing the number of training images does not necessarily improve the retrieval performance (Figure 1-c, d), while the general conclusions are difficult to draw regarding the effect of the number of training images on the retrieval performance.

## 5 Conclusion

We evaluated the small sample size (SSS) performance of evolutionary algorithms (EAs) for relevance feedback (RF) in image retrieval.

We compared four variants of EAs for RF — two of the existing ones, and the two new ones we introduced. Common for all variants is the hierarchical, region-based image similarity model, with region and feature weights as parameters. The difference between the variants is in the objective function of the EA used to adjust the model parameters. The objective functions included: (O-1) precision; (O-2) average rank; (O-3) ratio of within-class (i.e., positive images) and between-class (i.e., positive and negative images) scatter; and (O-4) combination of O-2 and O-3. We note that — unlike O-1 and O-2 — O-3 and O-4 are not used in any of the existing works dealing with EAs for RF.

The four variants of EAs for RF were evaluated on five test databases containing 61,895 general-purpose images, in 619 semantic categories. The comparison with the representative of the existing RF methods was performed as well.

The main contributions of the present work are: (1) *from the viewpoint of EAs for RF in image retrieval*: (a) we performed the first systematic evaluation of their SSS performance; (b) we introduced two new EA variants, shown to outperform the existing ones; (2) *from the viewpoint of RF in image retrieval in general*: through the comparison with the existing RF methods, we demonstrated that EAs are both effective and efficient approaches for SSS learning in region-based image retrieval.

## References

1. Brandt, S., Laaksonen, J., Oja, E.: Statistical shape features in content-based image retrieval. In: Proc. 15th Int. Conf. Pattern Recognition (ICPR-2000), Vol. 2. Barcelona, Spain (2000) 1066–1069
2. Chan, D. Y. M., King, I.: Weight assignment in dissimilarity function for Chinese cursive script character image retrieval using genetic algorithm. In: Proc. 4th Int. Workshop Information Retrieval with Asian Languages (IRAL'99). Taipei, Taiwan (1999) 55–62
3. Cordon, O., Herrera-Viedma, E., López-Pujalte, C., Luque, M., Zarco, C.: A review on the application of evolutionary computation to information retrieval. *International Journal of Approximate Reasoning*. 34(2-3) (2003) 241–264
4. Corel Corp.: Corel Gallery 3.0. <http://www3.corel.com/>. (2000)
5. Eiben, A. E., Schoenauer, M.: Evolutionary computing. *Information Processing Letters*. 82(1) (2002) 1–6
6. Korfhage, R. R.: Information Storage and Retrieval. John Wiley & Sons, Inc., New York, NY, USA (1997)
7. Laaksonen, J., Oja, E., Koskela, M., Brandt, S.: Analyzing low-level visual features using content-based image retrieval. In: Proc. 7th Int. Conf. Neural Information Processing (ICONIP'00). Taejon, Korea (2000) 1333–1338
8. Lew, M. S. (ed.): Principles of Visual Information Retrieval. Springer Verlag, London, UK (2001)
9. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs, 3rd edn. Springer-Verlag, Berlin, Germany (1996)
10. Porkaew, K., Chakrabarti, K., Mehrotra, S.: Query refinement for multimedia similarity retrieval in MARS. In: Proc. 7th ACM Int. Conf. Multimedia (MM'99). Orlando, Florida, USA (1999) 235–238
11. Rui, Y., Huang, T. S.: Relevance feedback techniques in image retrieval. In: [8], Ch. 9. (2001) 219–258
12. Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Analysis and Machine Intelligence*. 22(12) (2000) 1349–1380
13. Stejić, Z., Takama, Y., Hirota, K.: Genetic algorithm-based relevance feedback for image retrieval using Local Similarity Patterns. *Information Processing and Management*. 39(1) (2003) 1–23
14. Stricker, M., Orengo, M.: Similarity of color images. In: Proc. IS&T and SPIE Storage and Retrieval of Image and Video Databases III. San Jose, CA, USA (1995) 381–392
15. Wang, J. Z., Li, J., Wiederhold, G.: SIMPLiCity: Semantics-sensitive Integrated Matching for Picture Libraries. *IEEE Trans. Pattern Analysis and Machine Intelligence*. 23(9) (2001) 947–963
16. Zhou, X. S., Huang, T. S.: Small sample learning during multimedia retrieval using BiasMap. In: Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'01). Kauai, Hawaii, USA (2001) 11–17
17. Zhou, X. S., Huang, T. S.: Relevance feedback for image retrieval: a comprehensive review. *Multimedia Systems, Special Issue on Content-Based Image Retrieval (CBIR)*. 8(6) (2003) 536–544

# Co-retrieval: A Boosted Reranking Approach for Video Retrieval

Rong Yan and Alexander G. Hauptmann

School of Computer Science  
Carnegie Mellon University  
Pittsburgh PA, 15213, USA  
{yanrong, alex+}@cs.cmu.edu

**Abstract.** Video retrieval compares multimedia queries to a video collection in multiple dimensions and combines all the retrieval scores into a final ranking. Although text are the most reliable feature for video retrieval, features from other modalities can provide complementary information. This paper presents a reranking framework for video retrieval to augment retrieval based on text features with other evidence. We also propose a boosted reranking algorithm called Co-Retrieval, which combines a boosting type algorithm and a noisy label prediction scheme to automatically select the most useful weak hypotheses for different queries. The proposed approach is evaluated with queries and video from the 65-hour test collection of the 2003 NIST TRECVID evaluation.<sup>1</sup>

## 1 Introduction

The task of content-based video retrieval is to search a large amount of video for clips relevant to an information need expressed in form of multimodal queries. The queries may consist merely of text or also contain images, audio or video clips that must be matched against the video clips in the collection. Specifically this paper focuses on the content-based queries which attempt to find semantic contents in the video collection such as specific people, objects and events. To find relevant clips for content-based queries, our video retrieval system needs to go through the following steps as indicated in Figure 1. First, various sets of index features are extracted from the video library through analysis of multimedia sources. Each video clip (or shot) is then associated with a vector of individual retrieval scores (or ranking features) from different search modules, indicating the similarity of this clip to a specific aspect of the query. Finally, these individual retrieval scores are fused via a weighted linear aggregation algorithm to produce an overall ranked list of video clips.

It is of great interest to study the combination methods in the final step. This work considers approaches which rerank the retrieval output originally obtained from text features, using additional weak hypotheses generated from other

---

<sup>1</sup> This research is partially supported by Advanced Research and Development Activity (ARDA) under contract number MDA908-00-C-0037 and MDA904-02-C-0451.



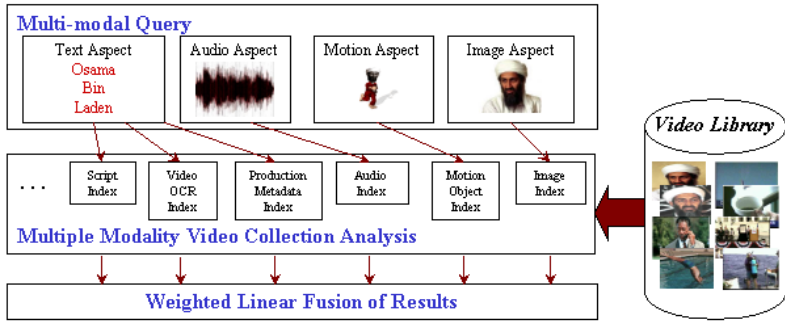


Fig. 1. Overview of our video retrieval system

modalities as evidence. Text retrieval first finds a set of relevant shots for each query, with associated scores that define an initial ranking. The selected weak hypotheses are then weighted and linearly combined in an attempt to improve upon the initial ranking.

Merely combining weak hypotheses with fixed weights or asking users to explicitly set weights are either inflexible or unrealistic. It is desired for the system to learn the linear weights automatically. Put in another way, the system should be able to pick out the most related concepts without feedback from users. For example, for the query "finding people on the beach" the ideal system can choose outdoors and people features to improve the initial retrieval. To achieve this, we apply a boosting type algorithm, which repeatedly learns weak classifiers from a reweighted distribution of the training data and combines the weak classifiers into one composite classifier. We also provide a noisy label prediction scheme which allows it to improve the initial ranking without any external training data. Experiments applied the proposed approach to a video test collection of over 65 hours from the 2003 content-based video retrieval track [1].

## 2 A Reranking Framework for Video Retrieval

Based on evidence from the best-performing video retrieval systems in the 2001 and 2002 NIST TREC Video Retrieval evaluation tasks, text features are demonstrated to be the most reliable ranking features for selecting semantically relevant shots in video retrieval. Text features span several dimensions including automatic speech recognition(ASR), closed captions(CC), video optical character recognition(VOCR) and production metadata such as published descriptions of the video. Typically a number of nearby shots are also retrieved since temporal proximity can somehow indicate content closeness. However, ranking of video shots cannot simply rely on these text features. One important reason is that words may be spoken in the transcript when no associated images are present, e.g. a news anchor might discuss a topic for which no video clips are available. A reporter may also speak about a topic with the relevant footage following later



**Fig. 2.** The key frames of top 8 retrieved shots for query "Finding Tomb at Arlington National Cemetery". These are retrieval results based on (a) text features (b) image features (c) text plus image features while filtering out news anchors and commercials

in the story, resulting in a major time offset between the word and the relevant video clips. As shown in figure 2(a), text retrieval will at times assign high scores to shots of studio settings or anchors which is usually not desirable. Moreover, word sense ambiguity may result in videos retrieved of the wrong meanings, e.g. either a river shore or a financial institution is possible for the word 'bank'. Speech recognition errors or VOCR errors may also result in incorrect retrieval. In general, retrieval using text features exhibits satisfactory recall but relatively low precision.

Fortunately, many complementary sources from various video modalities can be used to rerank the text retrieval output. These sources includes audio features, motion vectors, visual features (e.g. color, edge and texture histograms), and any number of pre-defined high-level semantic features (e.g. a face detector and an anchor detector). Generally speaking, none of these can fully capture the full content of the shots and therefore retrieval results based only on these features are mostly unsatisfying. Figure 2(b) depicts the top results of image-only retrieval which returns nothing related to the query. However, these weak features can provide some indication of how closely the video shots are related to the given specific query examples. They can also filter out irrelevant shots such as anchorpersons or commercials. Figure 2(c) illustrates the advantage of weak features which can augment text retrieval by finding the similar objects and filtering news anchor plus commercial shots.

These observations indicate that we should rely on text-based retrieval as the major source for answering semantic queries, while using the weak ranking functions from other modalities in combination to refine the ranking from the text retrieval. To be more general in the implementation, we convert the weak ranking features into a set of  $[-1,1]$ -valued weak hypotheses. Therefore, we propose the following re-ranking framework for video retrieval,

$$F(\mathbf{x}_i, \lambda) = \lambda_0 F_0(\mathbf{x}_i) + \sum_{t=1}^m \lambda_t h_t(\mathbf{x}_i) \quad (1)$$

where  $\lambda_t$  is the weight for the  $t^{th}$  ranking function,  $\mathbf{x}_i$  is the  $i^{th}$  video shot,  $F_0(\cdot)$  is the base ranking function generated by text retrieval and  $h_t(\cdot)$  is the output of the  $t^{th}$  weak hypothesis. Without loss of generality, we assume  $F_0(\mathbf{x}_i) = 1$  when shot  $\mathbf{x}_i$  is found to be relevant by text retrieval, otherwise  $F_0(\mathbf{x}_i) = 0$ .  $\lambda_0$  is typically set to be  $m \max_i(\lambda_i)$  after  $\lambda_i$  are learned. This allows  $F_0$  to dominate the retrieval results while the other weaker hypotheses  $h_t$  re-rank and adjust the output provided by  $F_0$ .

### 3 Co-retrieval: Boosted Reranking for Video Retrieval

In this section, we propose the Co-Retrieval algorithm which combines a boosting-type learning approach and a noisy label prediction scheme to estimate the weights  $\lambda$  without any external training data or user feedback.

#### 3.1 Boosted Reranking with Training Data

Let us begin with considering the case when a number of training data  $\{x_i, y_i\}$  are available for each query, where  $y_i \in \{-1, +1\}, i = 1..N$ . The goal of a learning algorithm is to find a setting for  $\lambda$  which leads to better retrieval outputs than  $F_0$ . Assuming all of  $x_i$  can be found by text retrieval, i.e.  $F_0(\mathbf{x}_i) = 1$ , we only need to optimize  $F(\mathbf{x}_i, \lambda) = \sum_{t=1}^m \lambda_t h_t(\mathbf{x}_i)$ . In the current setting, the learning algorithm will produce a function  $H : X \rightarrow \mathbf{R}$  which approximates the ordering encoded by the training data  $\{x_i, y_i\}$ . Formally, the boosting algorithm can be designed to minimize a loss function related to ranking misordering, i.e.  $\lambda = \arg \min_{\lambda} Loss(\{y_i, F(x_i, \lambda)\})$ . In analogy to classification problems, the loss function can be set to a monotonically decreasing function of margin  $y_i F(x_i, \lambda)$ , i.e.  $\sum_{i=1}^N L(y_i F(x_i, \lambda))$ . Two typical choices for function  $L$  are the exponential loss  $\exp(-x)$  used by AdaBoost and the logit loss  $\log(1 + \exp(-x))$  used by logistic regression. However, it has been argued that it is more reasonable to optimize ranking misordering through relative preferences rather than using an absolute labeling. Along this direction, the RankBoost algorithm proposed by Freund et al. [2] develops a boosting framework with ranking loss for combining multiple weak ranking functions. As a special case when the feedback function is *bipartite*, they provide an efficient implementation which actually minimizes the following ranking loss function, i.e.  $\sum_{y_i=+1} \sum_{y_j=-1} e^{-F(x_i, \lambda) + F(x_j, \lambda)}$ .

To handle different loss functions, we apply a unified boosting type algorithm for learning the combination of weak hypotheses. The boosting algorithm shown in Figure 3 are modified from the parallel update algorithm proposed by Collins et al.[3]. For each round  $k$ , the algorithm first updates the distribution  $q_{k,i}$  in a manner that increases the weights of examples which are misclassified. We will consider three loss functions in this step, i.e. exponential loss, logit loss and ranking loss. They have the following update rules respectively,

**Input:** Matrix  $\mathbf{M} \in [-1, 1]^{m \times n}$  where  $M_{ij} = y_i h_j(x_i)$  and  $\sum_{j=1}^n |M_{ij}| \leq 1$  for all  $i$ .

$N^+$  is the number of positive examples and  $N^-$  is the number of negative examples

**Output:**  $F(\mathbf{x}_i, \lambda) = \sum_{t=1}^m \lambda_t h_t(\mathbf{x}_i)$ , where  $\lambda_1, \dots, \lambda_m$  optimize  $\text{Loss}(\{y, F(x)\})$ .

**Algorithm:**

Let  $\lambda_1 = 0$ ,  $q_0 = (0.5, 0.5, \dots)$

For  $k = 1, 2, \dots$

1. Compute distribution  $q_k$  given  $\mathbf{M}$ ,  $\delta_k$  and  $q_{k-1}$
2. For every positive example  $\mathbf{x}_i$ , balance the distribution  $q_{k,i} = N^- q_{k,i} / N^+$
3. For  $j = 1, \dots, m$  :
 
$$W_{k,j}^+ = \sum_{i: \text{sign}(M_{ij})=+1} q_{k,i} |M_{ij}|$$

$$W_{k,j}^- = \sum_{i: \text{sign}(M_{ij})=-1} q_{k,i} |M_{ij}|$$

$$\delta_{k,j} = \frac{1}{2} \log(W_{k,j}^+ / W_{k,j}^-)$$
4. Update parameter:  $\lambda_{k+1} = \lambda_k + \delta_k$

**Fig. 3.** A unified boosting type algorithm with parallel-update optimization

$$q_{k+1,i} = \begin{cases} q_{k,i} \exp\left(-\sum_{j=1}^n \delta_{k,j} M_{i,j}\right) \\ q_{k,i} \left[ (1 - q_{t,i}) \exp\left(-\sum_{j=1}^n \delta_{k,j} M_{i,j}\right) + q_{k,i} \right]^{-1} \\ q_{k,i} \exp\left(-\sum_{j=1}^n \delta_{k,j} M_{i,j}\right) \sum_{y_l \neq y_i} q_{k,l} \exp\left(-\sum_{j=1}^n \delta_{k,j} M_{l,j}\right) \end{cases} \quad (2)$$

A new step, i.e. step 2, is added to balance the distribution between positive and negative examples. In fact many boosting algorithms take some reweighting or filtering approaches to obtain a balanced distribution such as the RankBoost.B[2], otherwise a constant -1 hypothesis is the likely output from all weak hypotheses. Finally the update vector is computed as shown in step 3 and added to the parameters  $\lambda_k$ . More details and the convergence proof can be found in [3].

The boosting algorithm requires access to a set of weak hypotheses  $h(\cdot)$  produced from ranking features. The most obvious choice for weak hypotheses is equal to the normalized ranking features  $f_i$ , i.e.,  $h(x) = a f_i(x) + b$  where  $a, b$  are constants to normalize  $h(x)$  to the range of  $[-1, +1]$ . If we only consider the relative ranking provided by the weak features instead of their absolute values, we can use the  $\{-1, 1\}$ -valued weak hypotheses  $h$  of the form, i.e.  $h(x) = 1$  if  $f_i(x) > \theta$  otherwise  $h(x) = -1$ , where  $\theta \in \mathbf{R}$  is some predefined threshold. In our implementation, we use the first definition for the features which compute the distance between given image/audio examples and video shots in the collection, e.g. the Euclidean distance from a color histogram. For the features generated by semantic detectors such as face and outdoors detectors, we choose the second definition because their relative ordering makes more sense than an absolute value. Rather than learning the threshold  $\theta$  automatically, we prefer to fix the threshold to 0.5 in terms of posterior probability.

**Table 1.** The number of relevant shots  $r_k$  in top  $k/n\%$  shots returned by text retrieval which is averaged over 25 TREC03 queries. The number in () is  $(r_k/r_n) * 100\%$

Top shots $k/n\%$	15%	25%	50%	75%	100%
$r_k$	184(32.11%)	248(43.8%)	367(64.1%)	487(84.7%)	573(100%)

### 3.2 Learning with Noisy Labels

So far we assume training data are available for the boosting algorithm, however, collecting training data for every possible query topic on the fly is not feasible in general. Alternative approaches have to be developed to generate a reasonable weight assignment without requiring a large human effort to collect training data. Formally, considering the entire set of shots returned by text retrieval  $\{x_1, \dots, x_i, \dots, x_n\}$ , we need to assign a set of (noisy) labels  $y_i$  which allows the boosting algorithm to improve the retrieval performance.

Without loss of generality, let us assume  $\{x_1, \dots, x_n\}$  are sorted in descending order of text retrieval scores and denote  $r_k$  the number of relevant shots in  $\{x_1, \dots, x_k\}$ . By analyzing the characteristics of text retrieval, we make the following assumption in the rest of this paper: The proportion of relevant shots in  $\{x_1, \dots, x_k\}$  is higher than in the entire set, i.e.  $r_k/k \geq r_n/n$ . In other words, the relevant shots are more likely to be higher ranked in the text retrieval. One partial explanation is that shots farther away from the content keyword location are lower ranked, with a lower probability of representing relevant concepts. Table 1 provides more insights to support our assumption. Therefore we can simply assign  $\{y_1, \dots, y_k\}$  as +1 and  $\{y_{k+1}, \dots, y_N\}$  as -1. In practice, we can augment the raw text retrieval scores with some highly accurate features to improve noisy label prediction, e.g. use anchor detectors to filter out irrelevant shots.

However, because automatically generated training data is quite noisy, regularization is generally required to reduce the effect of overfitting. Instead of introducing a penalty function into the loss function, we suggest two types of regularization approaches, 1. Use a  $\chi^2$  test to select features with confidence interval 0.1; 2. Set  $\lambda_t$  to be 0 if  $\lambda_t < 0$  for the nearest-neighbor-type features.

### 3.3 Related Work

Our approach builds on previous work which investigated the use of learning algorithms to improve ranking or retrieval. Collins et al. [4] considered a similar discriminative reranking approach to improve upon the initial ranking for natural language parsing. Tieu et al. [5] used boosting to choose a small number of features from millions of highly selective features for image retrieval. Blum et al. [6] proposed the co-training algorithm which trains a noise-tolerant learning algorithm using the noisy labels provided by another classifier.

In [7] we described a related co-retrieval approach which also attempted to learn the linear weights of different modalities with noisy label prediction. However, the current work represents several improvements over the previous

algorithm: (1) The proposed algorithm is more efficient because it only trains on the top video clips provided by text retrieval instead of the whole collection; (2) It applies a unified boosting algorithm to select the most useful weak features with different loss functions; (3) An additional regularization step is added to avoid overfitting; (4) The positive and negative distributions are balanced before training; (5) It converts ranking features into further weak hypotheses.

## 4 Experiments

Our experiments followed the guidelines for the manual search task in the 2003 NIST TRECVID Video Track 2003[1], which require an automatic system to search without human feedback for video shots relevant to 25 query topics in a 65-hour news video collection. The retrieval units were video shots defined by a common shot boundary reference. The evaluation results are reported in terms of the mean average precision(MAP) and precision at top  $N$  retrieved shots. We generated 7 weak ranking features in our experiments including 4 types of general semantic features (face, anchor, commercial, outdoors), and 3 types of image-based features generated by the Euclidean distance of color, texture and edge histograms when query image examples were available. Detailed descriptions on the feature generation can be found in [7].

The following experiments consider two typical scenarios for video retrieval: 1. when only keywords are provided we use only semantic ranking features; 2. when both keywords and image examples are provided we additionally use image ranking features. The co-retrieval algorithm works as follows: first return at most 400 video shots using text retrieval as a base ranking function, label top  $\alpha\%$  shots as positive and others as negative<sup>2</sup>, learn the parameter  $\lambda$  based on the noisy labels and feed this back to the reranking model. We set the number of rounds  $T$  to be 10000 and choose the best round using cross validation.  $\lambda_0$  is set to  $n|\max_t \lambda_t|$ , where  $n$  is number of weak hypotheses.

Figure 4 shows the performance improvement of Co-Retrieval without/with images examples over text retrieval alone. This improvement is achieved by successful reranking of top video shots. Table 2 lists a more detailed comparison for various retrieval approaches over mean average precision(MAP) at top 400 shots and precision at 10, 30 and 100 shots. Filtering out the anchor and commercial shots from text retrieval(**Text/A/C**) brings a slight performance improvement over text retrieval (**Text**). In contrast, Co-Retrieval with all three different loss functions (**CoRet+ExpLoss**, **LogLoss**, **RankLoss**) achieves a considerable and similar improvement over text retrieval in terms of all performance measures, especially when image examples are available MAP increases 5%. To investigate how noisy labels affect the results, we report the results of Co-Retrieval learning with truth labels(**CoRet+Truth**), which gives another 1.4% increase in MAP. This shows that the proposed algorithm is not greatly affected by the

<sup>2</sup> We augment noisy label prediction by reweighting shots identified as anchors or commercial from text retrieval scores.  $\alpha\%$  is simply set to 25%, because our experiments show that retrieval performance is not very sensitive to the choice of  $\alpha\%$ .



**Fig. 4.** The key frames of top 8 retrieved shots for query "Finding Tomb at Arlington National Cemetery". (a) Retrieval on text features (b) Co-Retrieval w/o image examples (c) Co-Retrieval with image examples

**Table 2.** Comparison between various retrieval approaches. See text for details

Approaches	Search w. Examples				Search w/o Examples			
	MAP	Prec10	Prec30	Prec100	MAP	Prec10	Prec30	Prec100
<b>Text</b>	0.157	0.292	0.225	0.137	0.157	0.292	0.225	0.137
<b>Text/A/C</b>	0.158	0.304	0.236	0.146	0.158	0.304	0.236	0.146
<b>Global Oracle</b>	0.188	0.368	0.259	0.16	0.164	0.336	0.235	0.152
<b>CoRet+ExpLoss</b>	0.206	0.444	0.307	0.171	0.177	0.352	0.261	0.156
<b>CoRet+LogLoss</b>	0.208	0.432	0.3	0.172	0.178	0.344	0.263	0.156
<b>CoRet+RankLoss</b>	0.207	0.448	0.301	0.172	0.178	0.344	0.26	0.156
<b>CoRet+Truth</b>	0.222	0.436	0.325	0.19	0.189	0.384	0.28	0.171
<b>Local Oracle</b>	0.285	0.512	0.344	0.199	0.212	0.436	0.304	0.171

overfitting problem typical with noisy labels. We also report the results of two oracles using the algorithms presented in [8]: An oracle of the single best combination weight for all queries (**Global Oracle**) and an oracle for the optimal combination weights per query (**Local Oracle**), which assumes all relevant shots are known ahead of time. This analysis shows that the Co-Retrieval consistently performs better than the theoretical optimal fixed-weight combination.

## 5 Discussions

*Why not optimize the performance criterion directly, that is mean average precision?* Table 2 shows that there is a considerable performance gap between the local oracle and Co-Retrieval even with true labels. Therefore it is of interest to ask if we can optimize the performance criterion directly. However these performance criteria are usually not differentiable and not convex, which leads to several problems such as local maxima, inefficiency and poor generalization. Table 3(a) demonstrates the fact that maximizing mean average precision with noisy labels is not generalized enough to boost the true mean average precision.

**Table 3.** Comparison between various retrieval approaches when image examples are available. (a) Co-Retrieval maximizing ExpLoss vs. MAP; (b) Co-Retrieval with regularization, without regularization and with automatically learned weak hypotheses

	MAP	Prec10	Prec30	Prec100
ExpLoss	0.206	0.444	0.307	0.171
MAP	0.182	0.376	0.265	0.16

(a)

	MAP	Prec10	Prec30	Prec100
Reg	0.206	0.444	0.307	0.171
NoReg	0.192	0.404	0.293	0.171
More $h$	0.171	0.34	0.236	0.142

(b)

*Why is boosting not overfitting?* It is well known that boosting type algorithms are not robust to noisy data and exhibit suboptimal generalization ability in the presence of noise, because it will concentrate more and more on the noisy data in each iteration[9]. However, our boosting algorithm does not seem to be affected by the overfitting problem even if our training data contains a lot of noise. Two answers come to mind. First, the regularization step improves generalizability which intentionally puts constraints on the choice of parameters. Table 3(b) compares the performance with and without regularization mentioned in Section 3.2 and shows that MAP will decrease about 1.4% without the regularization. Secondly, the version space of our weak hypotheses is much smaller than in most previous work such as [2], because we choose to fix the thresholds for weak hypotheses instead of learning these thresholds automatically. Table 3(b) shows how performance is much worse when the threshold is allowed to learn. To explain this, we utilize a theoretical analysis of boosting algorithms by Schapire et al.[9]. They claim that a bound on the generalization error  $P_{z \sim D}[\rho(z) \leq 0]$  depends on the VC-dimension  $d$  of the base hypothesis class and on the margin distribution of the training set. With probability at least  $1 - \delta$ , it satisfies,

$$P_{z \sim D}[\rho(z) \leq 0] \leq P_{z \sim D}[\rho(z) \leq \theta] + \mathcal{O} \left( \frac{1}{\sqrt{l}} \left( \frac{d \log^2(l/d)}{\theta^2} + \log(1/\delta) \right) \right).$$

This analysis supports our observation that the generalization error will increase when the VC-dimension  $d$  becomes higher or equally the hypothesis space becomes larger. Learning flexible thresholds allows the algorithms to achieve lower empirical results for noise-free labels, however, in the highly noisy case, reducing the hypothesis space turns out to be a better choice for learning.

## 6 Conclusions

This paper presents a reranking framework for video retrieval to augment retrieval based on text features. We also propose a boosted reranking algorithm called Co-Retrieval, which applies the boosting type algorithm to automatically select the most useful weak hypotheses for different queries. Our experiments on the TRECVID 2003 search task demonstrates the effectiveness of the proposed algorithms whether or not image examples are available. Finally, we discuss two



issues of Co-Retrieval on the choice of loss functions and the overfitting problem of boosting. As a possible extension, we can consider adding a relevance feedback function to the Co-Retrieval algorithm, which allows the interactive search system to rerank the current retrieval output given users' relevance feedback.

## References

1. TREC Video Track, "<http://www-nlpir.nist.gov/projects/tv2003/tv2003.html>," .
2. Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, "An efficient boosting algorithm for combining preferences," in *Proc. of ICML-98*, 1998, pp. 170–178.
3. M. Collins, R. E. Schapire, and Y. Singer, "Logistic regression, adaboost and bregman distances," in *COLT*, 2000, pp. 158–169.
4. M. Collins, "Discriminative reranking for natural language parsing," in *Proc. 17th Intl. Conf. on Machine Learning*, 2000, pp. 175–182.
5. K. Tieu and P. Viola, "Boosting image retrieval," in *Intl. Conf. on Computer Vision*, 2001, pp. 228–235.
6. A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *COLT*, 1998.
7. A. G. Hauptmann et al, "Informedia at trecvid 2003: Analyzing and searching broadcast news video," in *Proc. of (VIDEO) TREC 2003*, Gaithersburg, MD, 2003.
8. R. Yan and A. G. Hauptmann, "The combination limit of multimedia retrieval," in *Proc. of ACM Multimedia-03*, 2003.
9. G. Rätsch, T. Onoda, and K.-R. Müller, "Soft margins for AdaBoost," *Machine Learning*, vol. 42, no. 3, pp. 287–320, Mar. 2001.

# HMM Model Selection Issues for Soccer Video

Mark Baillie, Joemon M. Jose, and Cornelis J. van Rijsbergen

Department of Computing Science, University of Glasgow,  
17 Lilybank Gardens, Glasgow, G12 8QQ, UK  
{bailliem, jj, keith}@dcs.gla.ac.uk

**Abstract.** There has been a concerted effort from the Video Retrieval community to develop tools that automate the annotation process of Sports video. In this paper, we provide an in-depth investigation into three Hidden Markov Model (HMM) selection approaches. Where HMM, a popular indexing framework, is often applied in an ad hoc manner. We investigate what effect, if any, poor HMM selection can have on future indexing performance when classifying specific audio content. Audio is a rich source of information that can provide an effective alternative to high dimensional visual or motion based features. As a case study, we also illustrate how a superior HMM framework optimised using a Bayesian HMM selection strategy, can both segment and then classify Soccer video, yielding promising results.

## 1 Introduction

Live televised sporting events are now common place, especially with the arrival of dedicated digital channels. As a result, the volume of Sports video produced and broadcasted has increased considerably over recent years. Where such data is required to be archived for reuse, automatised indexing [2,3,5,8,9] is a viable alternative to the manual labour intensive procedures currently in practise. To date feasible solutions have not been developed. Current advancements, mainly the automatic identification of low level semantic structures, such as shot boundaries [3], semantic units [5,9] and genre classification [8] can reduce both the time and workload for manual annotation. Also, recognition of such low level structure is the basis for which further processing and indexing techniques can be developed. For example, labelling of low level segments can enable domain specific indexing tools such as exciting event detection [2] to be enhanced, utilising prior knowledge of content.

The difficulty with indexing Soccer video is that unrelated semantic components can contain visually very similar information, resulting in accuracy problems. For example, it is not uncommon for advertisements to display Sport sequences during televised events, to boost marketing appeal of a product, a potential source for error. However, audio is a rich, low dimension alternative to visual information that can provide an effective solution to this problem.

In this paper we model audio content using the Hidden Markov Model (HMM), a popular indexing framework. The main thrust of this research is

to provide an in-depth investigation into HMM model selection, where HMM is largely applied in an ad hoc manner for video content indexing [5,8,9]. We also investigate what effect poor selection can have on future indexing accuracy.

The remainder of this paper is structured as follows. In Section 2, we identify the potential factors that influence the application of a HMM. We then formally investigate three model selection strategies, in Section 3. As a case study, in Section 4, we illustrate how an extended HMM framework for segmentation and classification of Soccer video, can be optimised using model selection, yielding promising results. Finally, we conclude our work in Section 5.

## 2 Hidden Markov Model Issues

HMM is an effective tool for modelling time varying processes, belonging to a family of probabilistic graphical models able to capture the dynamic properties of temporal data [7]. Similar static representations, such as the Gaussian Mixture Model (GMM), do not model the temporal properties of audio data, hence the popularity of HMM in the fields of Speech Recognition [4,7], temporal data clustering [6,7] and more recently Video Retrieval [2,3,5,8,9]. An important issue when employing a continuous density HMM framework is model selection [6,4,7]. For example, a crucial decision is the selection of both an appropriate number of hidden states and (Gaussian) mixture density estimation per state. Accurate segmentation and classification is dependent on optimal selection of both these parameters. An insufficient number of hidden states will not capture enough detail, such as data structure, variability and common noise, thus losing vital information required for discrimination between groups. A greater number of hidden states would encapsulate more content, though precise and consistent parameter estimation is often limited by the size and quality of the training data. As the number of parameters increase, so does the number of training samples required for accurate estimation. Larger more enriched models require a greater volume of training data for precise parameter estimation. A further problem with complex models is overfitting. HMMs, specifically designed to discriminate between content, can become too detailed and begin to mirror nuances found in unrelated groups, deteriorating classification accuracy.

HMM application for Video Retrieval has so far been ad hoc, with little investigation into model selection and the potential side effects on system performance. In the literature, a common theme is to apply domain knowledge or intuition for HMM model selection. Such application includes shot boundary detection [3], news video segmentation and classification [5], TV genre labelling [8] and ‘Play’ or ‘Break’ segmentation [9] for Soccer video. This strategy can be helpful when matching a known number of potential states found in the data, such as shot segmentation [3]. However, there has been little research into how suitable this strategy is when applied to broad content classes found in video. For example, Wang et. al. [8] employ the same number of hidden Markov states for modelling entire video genre such as Sport and News, ignoring differences in the underlying structure found in each separate domain.

Eickeler et. al. [5], apply domain knowledge to News Broadcasts, building a superior HMM based on a preconceived topology. Each state of a superior HMM is represented by a simple HMM that models a broad content class found in News video. However, there is no investigation into model selection for these simple HMMs. Xie et al [9] segment and classify ‘Play’ and ‘Break’ segments for Soccer video, by using HMMs to model motion and colour distribution statistics. ‘Play’ segments correspond to camera shots that track the flow of the game. To model both segments, the authors use a series of simple HMM models, with a varying number of hidden states. For segmentation and classification, the output from each model is then combined using a dynamic programming (DP) algorithm, itself a first order Markov process. In fact, this application ignores the temporal properties of the HMM, suggesting a simpler classifier such as the Gaussian Mixture Model, applied in conjunction with the DP algorithm, may be as effective.

### 3 HMM Model Selection

The main goal of model selection is to choose the simplest possible model without a deterioration in performance. This is especially important given the difficulty and practicality of generating large, varied training sets. In this Section, we investigate three model selection strategies and what effect each has on classification performance. The three selection strategies are: an exhaustive search approach, the Bayesian Information Criterion (BIC) [4,6] and the Akaike Information Criterion (AIC) [1] (formulae can be found in references). Exhaustive search, a simple linear search algorithm, involves training and testing a series of HMMs, where the parameter in question is iteratively increased until a stopping threshold is reached. For each iteration, the predictive likelihood of a HMM generating a test sample is calculated, also known as the out of sample log-likelihood. Using a stopping criteria on the predictive likelihood score is important. For example, increasing the number of states will also increase the predictive likelihood until each training sample is eventually modeled by its own unique hidden state.

The two remaining strategies are BIC and AIC, both popular in the Statistical literature. Each strategy penalises the predictive likelihood with a term that is derived from the number of parameters in the model. The major difference between approaches, is the derivation of this penalty term. The penalty term for AIC, only accounts for the number of free parameters in the HMM, while the BIC penalty term also factors in the amount of training data available. Smaller training samples will generate larger penalty scores, hence the advantage in predictive likelihood found with more complex models is eventually outweighed by this penalty term. We then assume the optimal model is found at the maximum predictive likelihood score, avoiding the need to threshold.

#### 3.1 Data Set

To evaluate each strategy, we generated a data set of 12 games ranging between 2 to 3 hours in length. We manually labelled the audio into three main semantic content classes found in Soccer video; ‘Game’, ‘Studio’ and ‘Advertisement’.

‘Studio’ segments contain an introduction plus pre and post match discussion and analysis of the live game, usually set inside a controlled soundproof studio. ‘Game’ segments consist of the live match, where the soundtrack contains a mixture of both commentary and vocal crowd reaction, alongside other environmental sound such as whistles, drums and clapping. ‘Advert’ segments can be identified by the almost chaotic mixture of highly produced music, voice and sound effects. Segmentation and labelling of these low level segments is beneficial, especially for reducing indexing errors during higher level tasks. For example, identifying the boundaries of a ‘Game’ segment is vital before event detection [2]. A decrease in precision would occur if the data was not pre-segmented and labelled. Similar information from unrelated content such as music or sound effects, can then be wrongly identified as a key event.

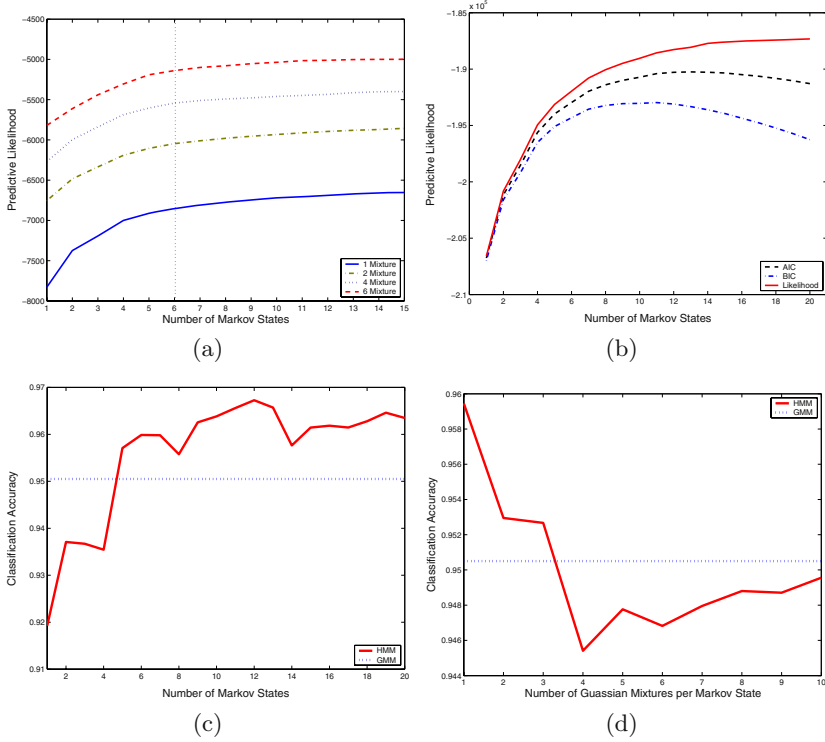
### 3.2 Number of Markov States Selection

A series of HMMs were implemented, modelling the ‘Game’, ‘Studio’ and ‘Advert’ content classes. The audio stream for each file was parameterised using 14 Mel-Frequency Cepstral coefficients (MFCC) with an additional Log Energy measurement [7]. MFCC coefficients are specifically designed and proven to characterise speech well. MFCC has also shown to be both robust to noise and useful in discriminating between speech and other sound classes [2,4].

For each class, a series of ergodic, continuous density HMMs [7] with increasing number of states ranging from 1 to 20, were iteratively implemented. Each model was first generated from a training sample, then the predictive likelihood score was calculated on a separate test set. Both labeled data samples were generated from the same pool of data and after one complete run, each sample was randomly changed. This was repeated 15 times to achieve a true reflection of the model generation process, limiting the effect of unusually good or bad runs.

Importantly, each HMM was assigned a singular Gaussian density per hidden state. An informal investigation using synthetic data, indicated that it was more important to identify the correct number of states first, to avoid searching through an unnecessary large number of hidden state and mixture combinations. For example, 100 HMMs of different state and mixture combination were implemented using data generated by a 6 state HMM, with 6 mixtures per state, Figure 1(a). Using the exhaustive search approach, increasing the number of mixtures did not effect correct hidden state number identification. As a result, we could first identify the optimal number of hidden Markov states for each content class, implementing a singular density function per state. Then in a separate step, the optimal number of Gaussian density components could be found, reducing the number of parameter combinations to be implemented.

Using the ‘Game’ class as an example, Figure 1(b) displays the mean of the 15 initialisations, for all selection strategies. For the exhaustive search approach, the predictive likelihood increases as a new hidden Markov state is added to the model. There is a rapid rise that levels off between 14 to 20 states, suggesting the model was beginning to overfit the training data. A stopping threshold, empirically determined using synthetically generated data, Figure 1(a), was reached



**Fig. 1.** (a) The predictive likelihood scores for HMMs with increasing state and mixture component number. (b) A comparison of model selection strategies for hidden state selection. Notice, both the AIC and BIC scores peak, while the predictive likelihood score continues to increase. (c) Classification accuracy versus number of hidden Markov states. (d) Classification accuracy versus the number of Gaussian mixture components.

when adding a 14<sup>th</sup> state. For the BIC strategy, the predictive likelihood also increased dramatically but peaked and then tailed off. The maximum BIC score was found to be 9 states. For the AIC strategy, a similar pattern occurred, where the maximum AIC score was found at 12 states. There was a similar trend for the remaining two content groups. BIC selected the simplest model followed by AIC, then the exhaustive search method.

We also evaluated what effect iteratively adding hidden Markov states had on classification accuracy, Figure 1(c). As a comparison, the simpler GMM classifier [7], which does not model time as a property, was used as a baseline. The mean classification accuracy gradually increased as new hidden states were added to the HMM. After the 5<sup>th</sup> hidden state was added, the HMM began to outperform the GMM classifier. A 12 state HMM was found to be optimal for this content class, the same model selected using the AIC strategy. A similar pattern emerged for the remaining content classes. An improvement in classifi-

cation accuracy over the baseline GMM was recorded, when a certain number of states were added to the HMM.

### 3.3 Number of Gaussian Mixtures per Markov State

The same implementation issues arise with the selection of mixture components that model the emission densities per hidden Markov state. For example, speech recognition systems have identified that HMMs with multiple Gaussian mixtures perform better than those with a singular density function [4,7]. A mixture of Gaussian components can model the multi-modal emission densities that represent variation found in speech. However, selecting too many mixture components can result in overfitting. Thus, we repeated the previous experiment, this time implementing HMMs with increasing Gaussian mixture components per state.

For each strategy, each content class was modeled with mixtures ranging from 1 up to 10, fixing each HMM with the optimal number of hidden states identified in the previous section. For example, for one content class, 3 HMMs were implemented using the optimal number of states identified by each selection strategy. To limit overfitting further. The covariance matrices were constrained to be diagonal for each individual mixture, reducing the number of free parameters. Each model setting was initialised 15 times, changing the data samples randomly after a complete run. Our findings again indicated that the BIC strategy selected the simplest model followed by AIC. The exhaustive search strategy again selected the more complex HMMs.

We also analysed what effect iteratively adding Gaussian mixtures per model had on classification accuracy, Figure 1(d). From our results, we discovered a decrease in classification accuracy as mixtures were added to a singular density HMM. This trend was consistent across all strategies and for all content classes. Figure 1(d), is an illustration of a 9 state HMM for the ‘Game’ class, as the number of mixture components is iteratively increased. Classification accuracy decreases until 4 states are added, with a small reverse in trend afterwards. After three mixtures, the model performance became poorer than that of the GMM. This result was mirrored across the remaining two content classes and could be indicative of both poor parameter estimation given increased model complexity, as well as overfitting. To summarise. A singular density HMM produced the best classification accuracy when compared to the same HMM with multiple mixture components.

### 3.4 Optimal HMM Model Evaluation Experiment

In the previous section, we identified 3 optimal HMMs for each content class, using three selection strategies. Next, these HMMs were formally compared over a new test set, using a baseline GMM classifier for comparison. The test set was approximately 2 hours in length, divided into 10 second observation sequences, labelled into each content class. For all content classes, a HMM was first generated from the labeled data used in the previous section. The HMM was then tested on the new sample. For each strategy, each new individual sequence was

**Table 1.** Confusion matrix. The % of correctly classified observations are in bold.

Classification (%)														
Correct Class	Game				Studio				Advert				Total	
	LIK	BIC	AIC	GMM	LIK	BIC	AIC	GMM	LIK	BIC	AIC	GMM		
Game	<b>89.6</b>	<b>92.7</b>	<b>90.4</b>	<b>90.4</b>	1.8	1.1	1.0	2.9	8.5	6.2	8.6	6.7	100%	
Stud	4.6	5.2	5.1	2.9	<b>89.1</b>	<b>86.8</b>	<b>87.6</b>	<b>90.3</b>	6.2	8.0	7.2	6.8	100%	
Advt	1.1	1.0	1.1	1.4	3.5	3.4	3.0	3.7	<b>95.4</b>	<b>95.6</b>	<b>95.9</b>	<b>94.9</b>	100%	

assigned to the content class that produced the highest HMM likelihood score, found using the Viterbi decoding algorithm [7].

The results in Table 1, indicated no significant difference in terms of classification accuracy across all selection strategies, and across each content class. Overall, the ‘Studio’ classifier indicated the worst performance, where the majority of false classifications were samples with speech containing background noise, wrongly labelled as ‘Game’ or ‘Advert’. False classification from the ‘Game’ class again included sequences containing speech. These observations contained little or no environmental sound associated with the ‘Game’ class, resulting in misclassification. Samples containing music played inside the stadium, or other peculiarities such as tannoy announcements, were also wrongly labelled into the ‘Advert’ class. These sound events reflected similar content found in the ‘Advert’ class. The ‘Advert’ HMM produced the highest classification accuracy for all selection methods, where the majority of false classifications were labeled into the ‘Studio’ category. These errors were typically clean speech samples.

Given that the BIC selection criterion chose the simplest HMMs overall, there was no obvious detriment in performance. In fact the HMM selected by BIC for the ‘Game’ class, produced the highest classification accuracy. However, the same selection strategy resulted in the lowest classification accuracy for the ‘Studio’ group. Interestingly, for the same content class the baseline GMM classifier recorded the best result. In fact, across all content classes, the GMM displayed comparable results when compared to the HMM.

### 3.5 Discussion

From experimentation, we illustrated the importance of model selection, where a gain in performance can be found when selecting HMMs methodically. For many Video indexing applications of HMM, this type of approach is not adopted [5, 8,9], highlighting optimisation issues for each system. Selecting too few or too many hidden states can produce poor classification performance, as shown from the experimentation of three model selection techniques.

The BIC method selected the simplest HMMs without significantly decreasing classification accuracy. In some cases, displaying a higher classification accuracy than more complex HMMs. Also, the BIC strategy has an obvious advantage over an exhaustive search approach. The BIC penalty term creates a maximum peak in the predictive likelihood score. We assume this maxima to be the optimal



HMM. Hence, to find an optimal solution. The number of HMMs required to be implemented can be reduced by avoiding an iterative addition of parameters. For example, a bisection search strategy such as a Newton-Raphson could be implemented to find the maximum BIC score.

From experimentation, an important discovery was the effect increasing the number of mixture components had on classification accuracy. Adding further Gaussian mixtures to a singular density HMM, created a detrimental effect. Increasing the complexity resulted in poor parameter estimation and overfitting. In most cases, after two or more mixtures were added, the baseline GMM recorded better results. In fact, for the task of classification, the HMM framework did not perform significantly better than the GMM overall. For this problem, GMM has been shown to be as effective when compared to the more complex HMM.

## 4 A Segmentation and Classification System

In this section, our aim is to segment and then classify Soccer video files using audio information alone. We present a case study, illustrating how optimally selected HMMs using BIC, can be integrated into a superior HMM framework [5]. This combination scheme utilises both domain knowledge as well as statistical model selection, where each optimised HMM becomes a single state in a unified HMM topology. This superior HMM allows for an entire video file to be segmented, classifying semantic segment units in a single pass. The advantage of applying this decision process is the ability to incorporate the temporal flow of the Video into the segmentation process, limiting error. For example, restricting movement from the ‘Advert’ to ‘Game’ segments can be mirrored in the state transition matrix in the superior HMM. Also, an input and output state, to note the beginning and end of each video file are included.

To evaluate this technique, given the limited data set, we applied a ‘leave one out cross validation’. 11 complete video files were used for model training. The ‘held’ out video was then used to evaluate the superior HMM. This procedure was repeated, holding out each video in turn, until segmentation and classification was achieved for all videos in the data set. We indexed all 12 video files using the Viterbi decoding algorithm, where each one second is assigned to a state in the superior HMM that represented a specific content class. An ambiguity window of 2 seconds was allowed for each segment change when comparing the indexed files with the manually generated truth data. This was to limit small alignment errors between the ground truth and the model output.

The majority of segment boundaries were identified with 95.7% recall and 89.2% precision. 97.9% of the segments were correctly labeled. Even allowing for the ambiguity window. Those segment changes that were not picked up correctly were largely due to alignment errors, where the detected boundary was missed by a few seconds. False detections for segment change mostly involved wrongly identified segment transition between ‘Studio’ to ‘Game’ segments or vice versa. For example, false boundary changes were marked during a ‘Game’ segment where there was a decrease in crowd sound. A simple solution to this

problem would be to add a state duration element into the HMM framework. One complete ‘Game’ segment spans approximately 45 minutes. Incorporating a time distribution could avoid false classifications, especially during quiet spells in a ‘Game’ segment.

## 5 Conclusions and Future Work

In this paper, we investigated three HMM model selection strategies, examining factors that can effect the application of a HMM framework. We found the BIC selection strategy to be the most effective. By then incorporating optimal HMMs into a unified framework, we then illustrated how a superior HMM can be applied to both segment and classify the low level structure of Soccer video, yielding promising results. Labeling was achieved by modelling underlying audio patterns found in each semantic unit.

Intended future work will include the extension of the superior HMM framework to include visual, motion and textual information sources. Another active area of interest will be incorporating the classification of smaller sub-groups such as crowd cheering for event detection [2], music and speech. Thus extending the HMM framework to include a more complete topology for the annotation of live Soccer broadcasts. Finally, we wish to compare this system against other frameworks, a requirement highlighted during experimentation.

**Acknowledgements.** The authors would like to thank Prof. Mark Girolami and Vassilis Plachouras.

## References

1. H. Akaike. A new look at the statistical model identification. In *Trans. Automatic Control*, volume AC-19, pages 716–723. IEEE, Dec 1974.
2. M. Baillie and J. M. Jose. Audio-based event detection for sports video. In *CIVR2003*, pages 300–310, IL, USA, July, 2003.
3. J. S. Boreczky and L. D. Wilcox. A hidden markov model framework for video segmentation using audio and image features. In *ICASSP*, pages 3741–3744, Seattle, May 1998. IEEE.
4. S. S. Chen and R. A. Gopinath. Model selection in acoustic modeling. In *Proceedings of Eurospeech-99*, Hungary, September 1999. Eurospeech.
5. S. Eickeler and S. Muller. Content-based video indexing of tv broadcast news using hidden markov models. In *ICASSP*, Phoneix, USA, 1999. IEEE.
6. C. Li and G. Biswas. A bayesian approach to temporal data clustering using hidden markov models. In *ICML*, pages 543–550, California, 2000.
7. L. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, Englewood Cliffs, NJ, USA, 1993.
8. Y. Wang, Z. Liu, and J. Huang. Multimedia content analysis using both audio and visual clues. In *IEEE Signal Processing Magazine*. IEEE, 2000.
9. L. Xie, S.-F. Chang, A. Divakaran, and H. Sun. Structure analysis of soccer video with hidden markov models. In *ICASSP*, 2002.

# Tennis Video Analysis Based on Transformed Motion Vectors

Peng Wang<sup>\*1</sup>, Rui Cai<sup>1</sup>, and Shi-Qiang Yang<sup>2</sup>

<sup>1</sup> Department of Computer Science and Technology  
Tsinghua University, Beijing, 100084, China  
{wangp01, cairui01}@mails.tsinghua.edu.cn

<sup>2</sup> Department of Computer Science and Technology  
Tsinghua University, Beijing, 100084, China  
yangshq@tsinghua.edu.cn

**Abstract.** Motion Vectors (MV) indicate the motion characteristics between two video frames, and has been widely used in the content-based sports video analysis. Previous works on sports video analysis have proved the effectiveness and efficiency of the MV-based methods. However, in the tennis video, the MV-based methods are seldom applied because the motion represented by MV is greatly deformed relative to the player's true movement due to the camera's diagonal shooting. In this paper, an algorithm of MV transformation is proposed to revise the deformed MV using a pinhole camera model. With the transformed MVs, we generate the temporal feature curves and employ Hidden Markov Models to classify two types of player's basic actions. Evaluation on four hours live tennis videos shows very encouraging results.

## 1 Introduction

As one of the most salient visual characteristics, motion feature is widely used in the content-based sports video analysis. In MPEG stream, Motion Vector (MV), extracted from the compressed video stream, reflects the displacement of a macro block, and most of current motion features employed in sports video analysis ground on MV. Duan et al. [1] give a comprehensive summarization of MV-based mid-level representations and corresponding implementations for sports game analysis. Also based on MV, Ma [2] calculates the motion energy spectrum for video retrieval, and in [3] the motion energy redistribution function is proposed for semantic event recognition. It shows that the MV-based methods have efficient computation and effectual performance for most generic applications.

However, the MV-based methods are seldom utilized in the tennis video analysis. Conventional methods on tennis video analysis mainly focus on detecting and tracking of player or ball in the image sequence [4] [5], as well as incorporating with human gesture and behavior analysis [6]. Although these computer

---

<sup>\*</sup> This work was supported by the *National High Technology Development 863 Program* of China and the *National Grand Fundamental Research 973 Program* of China

vision related methods may provide the elaborate annotation of tennis game, they have complicated implementation, inflexible utilization and nontrivial limitation. With our investigation, the main reason baffling the utilization of the MV-based methods in tennis video is that the camera is diagonal located but not perpendicular to the tennis court plane. And thus, the motion vector estimated from the tennis video can not correctly represents the player's true movement. The magnitude of the MV is reduced and the orientation of MV is distorted. The deformation is particularly notable for the player at the top half court. To utilize the MV-based methods in tennis video analysis, the revision of MV must be resolved according to the player's true movement.

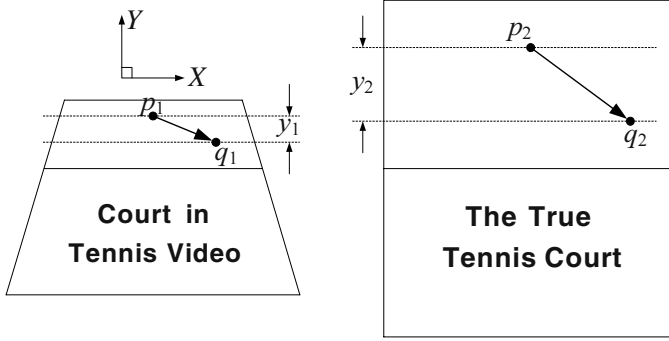
In this paper, an algorithm of motion vector transformation for tennis video analysis is proposed, in which the transformation is implemented based on the pinhole camera model. In this algorithm, the tennis court and net lines are detected to construct the pinhole camera model. Two types of player's basic actions are classified to verify the proposed algorithm. In the classification, the temporal motion curves are generated and the HMM-based classifiers are built using our previous work[3]. Experiments on live tennis videos demonstrate the effectiveness and efficiency of the proposed algorithm of motion vector transformation.

The rest of this paper is organized as follows. Section 2 presents the motion vector transformation by utilizing the pinhole camera model. In Section 3, the implementation of the transformation for classifying player's basic actions is introduced. Then, experiments and discussion are given in Section 4. Finally, Section 5 presents the conclusion and future works.

## 2 Motion Vector Transformation

The camera in tennis game is usually placed right above the vertical symmetrical axis of the tennis court, and thus the rectangle court is transferred to an isosceles trapezoid court, as shown in Fig. 1. Consequently, the player's movement in tennis game is also distorted and the MV estimated from video sequence can not represent the true movement. As illustrated in the left of Fig. 1, a motion vector can be denoted as the displacement from a given point  $p_1$  in the current frame to its corresponding point  $q_1$  in the next frame. Supposing the player is watched moving from  $p_1$  to  $q_1$  in video sequence, the true movement should be from  $p_2$  to  $q_2$ , as shown in the right of Fig. 1. Comparing with the true motion, the magnitude of the estimated motion in video sequence is reduced and the orientation is also distorted. The distortion in the vertical direction is especially prominent, and there is always  $y_1 < y_2$  in Fig. 1. Such deformation makes it difficult to analyze player's true movement based on the original MVs, for instance, we can hardly tell whether the player is taking the net or not just depending on the vertical projection of the MVs. However, this task would become feasible if the original MVs can be revised according to the actual movements.

In fact, for points  $p_1$  and  $q_1$ , if the corresponding points  $p_2$  and  $q_2$  in tennis court plane can be correctly located, the transformation will be achieved, i.e.



**Fig. 1.** Illustration of the motion vector deformation in tennis video

$$\begin{cases} \mathbf{MV}_{original} = q_1 - p_1 \\ \mathbf{MV}_{transformed} = q_2 - p_2 \end{cases} \quad (1)$$

Thus the essential problem is that for any given point in video frame, how to find the corresponding point in tennis court plane. To perform this task, a pinhole camera model is employed as illustrated in the top left of Fig. 2. For a pinhole camera, there is

$$L/l = u/f \quad (2)$$

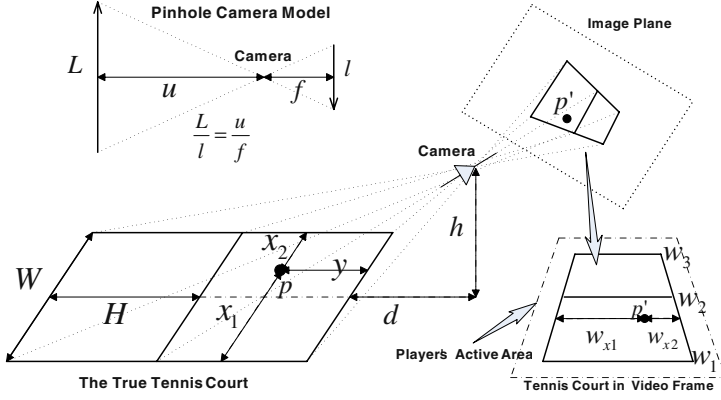
where  $u$  and  $f$  denote the object distance and the camera focus,  $L$  and  $l$  denote the lengths of object and image respectively. Supposing the horizontal distance between the camera and the bottom baseline of the true tennis court is  $d$ , and the height of camera from the ground is  $h$ , with Eq. (2), there are

$$\begin{cases} W/w_1 = \sqrt{d^2 + h^2}/f \\ W/w_2 = \sqrt{(d+H)^2 + h^2}/f \\ W/w_3 = \sqrt{(d+2 \cdot H)^2 + h^2}/f \end{cases} \quad (3)$$

Here  $W$  and  $H$  denote the width and half height of the true tennis court [8], and  $w_1$ ,  $w_2$ ,  $w_3$  respectively represent the lengths of the bottom baseline, net line and top baseline in the image plane, as shown in Fig. 2.

For any given point  $p'$  in the trapezoidal court in image plane, the line passing through  $p'$  and being parallel with the baselines is segmented by  $p'$  and the two court sidelines into two parts, whose lengths are denoted as  $w_{x1}$  and  $w_{x2}$  respectively. The position of  $p'$  is uniquely represented by  $w_{x1}$  and  $w_{x2}$ . Supposing  $p$  is the corresponding point in the true tennis court of  $p'$ ,  $p$  is uniquely represented by  $x_1$ ,  $x_2$ ,  $y$  which denote the distances between  $p$  and the two sidelines and bottom baseline, as illustrated in Fig. 2. With the pinhole camera model in Eq. (2), the relations between  $(x_1, x_2, y)$  and  $(w_{x1}, w_{x2})$  are

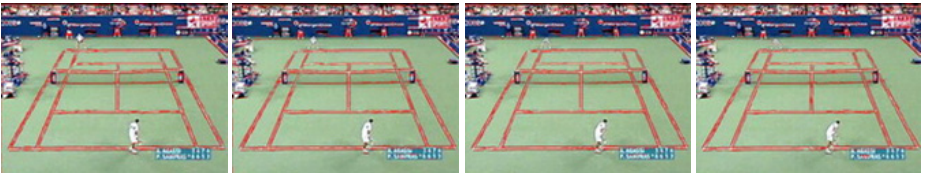
$$\begin{cases} W/(w_{x1} + w_{x2}) = \sqrt{(y+d)^2 + h^2}/f \\ w_{x1}/w_{x2} = x_1/x_2 \quad \text{and} \quad x_1 + x_2 = W \end{cases} \quad (4)$$



**Fig. 2.** Pinhole camera model based transformation between the image plane and the true tennis court plane

With Eq. (3), the parameters of  $d$  and  $h$  can be solved, thus for a given point in video frame (the image plane), the position of the corresponding point in the true tennis court plane can be calculated with Eq. (4). Using the point transformation functions, the two end points of a MV are transformed to the true tennis court plane, and the new motion vector is calculated by taking the difference between the two transformed end points, as shown in Eq. (1).

In most of the *Game Shots* of tennis video, the camera is usually appropriately placed and our assumption is approximately justified. With the robust line detection algorithm proposed in [7], the exact position of the trapezoidal court in tennis video, including the lengths of the borders and the coordinates of the corners, can be obtained through averaging the line detection results in several beginning frames of the *Game Shots*. Fig. 3 gives an example of the court line detection results in several consecutive video frames. The tennis court and net line in image are identified with the red lines. It is shown that the performance of the line detection algorithm is reliable in practice.



**Fig. 3.** Results of tennis court line detection in consecutive video frames

When the position information of the trapezoidal tennis court is obtained, all motion vectors in the *Player Active Area* are transformed to the true tennis court plane. The *Player Active Area* is defined as a larger isosceles trapezoid

covering the tennis court in video frame, as the trapezoid in dash-dot line shown in the right bottom of Fig. 2.

### 3 Classification of Player's Basic Actions

In order to evaluate the performance of the motion vector transformation, we apply it to the semantic analysis of tennis video. A tennis video generally includes various scenes, but the most important shots are those *Game Shots*. In this paper, the *Game Shots* have been selected from the video sequence by the color-based approach in [4]. Two types of player's basic actions are considered in current experiments: *Net Game* and *Baseline Game*. *Net Game* is that the player moves into the forecourt and toward the net to hit volleys, and *Baseline Game* is that the player hits the ball from near the baseline [8].

#### 3.1 System Overview

The system overview for classifying the player's basic actions is illustrated in Fig. 4. For each *Game Shot*, the *Original MVs* are firstly extracted from the video stream, and then are fed into the proposed *Transformation Algorithm* to calculate the *Transformed MVs*. As introduced in Section 2, with the *Line Detection* algorithm, the position information of *Tennis Court* is identified to build up the *Transformation Algorithm* for the given *Game Shot*.

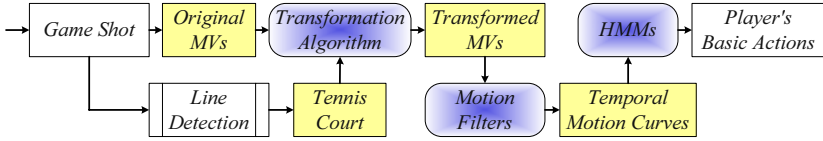
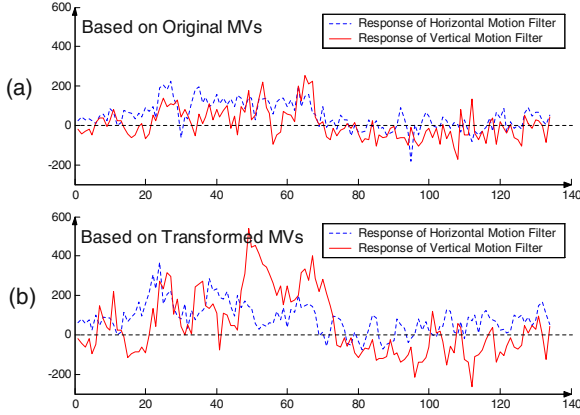


Fig. 4. System overview for classification of *Players' Basic Actions*

Subsequently, based on the *Transformed MVs*, the method proposed in our previous work [3], which is validated in sports game event recognition, is employed to classify *Player's Basic Actions*. As described in [3], energy redistribution measurement and weight matrix convolution are implemented on motion vector fields. First, the energy redistribution function provides a way to convert motion vector field to energy matrix, and then the weight matrix as *Motion Filter* is designed to detect the response of particular motion patterns. With the *Motion Filter*, the temporal multi-dimensional motion vector fields become a one-dimensional motion response curve called *Temporal Motion Curve*. More details can be found in [3]. In this paper, the horizontal and vertical motion filters are designed and two *Temporal Motion Curves* are generated to characterize the horizontal and vertical motions within the *Game Shot*. These curves as features are then used to classify the *Player's Basic Actions* by using *Hidden Markov Models*, which will be detailedly introduced in the next subsection.

For classifying the basic actions of players in top half court and bottom half court respectively, the *Original MV* and the *Transformed MV* are divided into two parts with the detected net line. For comparison, two *Temporal Motion Curves* are respectively calculated based on the *Original MVs* and the *Transformed MVs*. Fig. 5 shows an example of the two *Temporal Motion Curves* of *Net Game* in the top half court. The  $X$  axis denotes the frame number and the  $Y$  axis denotes the calculated motion response value. Positive value on the vertical motion curve means movement to the net line, and movement to the right sideline for the horizontal motion curve. From frame 1 to 80, the player runs to take the net from the left end of the baseline toward the right end of the net, then from frame 81 to 134, the player walks back from the net to the right end of the baseline. As shown in Fig. 5 (a), both motion curves are quite noisy, and the vertical motion curve is too irregular to characterize the net approach movement. In Fig. 5 (b), the responses of horizontal and vertical motion filters are both enlarged, and the segment representing the net approach is more evident.



**Fig. 5.** Comparison between *Temporal Motion Curves* built on (a) the original MVs and (b) the transformed MVs respectively

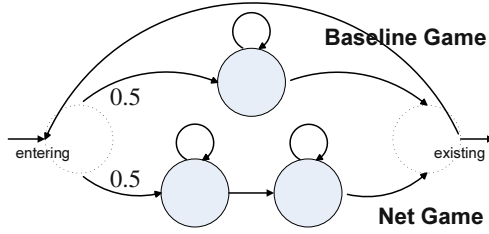
### 3.2 HMM-Based Action Classification

For player in certain half court, two HMMs for the net-game segment and baseline-game segment within a *Game Shot* are respectively built, as shown in Fig. 6. The baseline-game segment is modeled with a one-state HMM, with the continuous mixture of Gaussians modeling the observations. The component number of Gaussian mixture is selected as three, since the movements in baseline game are mainly composed of left, right and keep still. The HMM for net-game segment is a two-state left-to-right model, one state for approaching the net, and the other for returning the baseline. Each state is also modeled with a three-component continuous mixture of Gaussians. The observation vector for both



HMMs is of four dimensions, i.e. the response values of horizontal and vertical motion filters and their gradient values.

As illustrated in Fig. 6, the two HMMs are then circularly connected to a higher-level model with an entering and an existing state, which represents the transition between net-game segments and baseline-game segments within a *Game Shot*. The transition probabilities from the entering state to the two sub-HMMs are both set to 0.5. The HMM model parameters are trained using the EM algorithm. Training data are manually chopped into homogeneous net-game/baseline-game chunks; EM for the net-game models is conducted over every complete net-game chunks, and vice versa for baseline-game models. In recognition phase, viterbi algorithm [9] is applied to find the global optimal state path given a observation sequence.



**Fig. 6.** HMMs for classification of *Player's Basic Actions*

## 4 Experiments

Four hours recorded live tennis videos are used in current experiments to validate the performance of the proposed algorithm. The experimental video data are collected from the matches of A. Agassi and P. Sampras at US Open 2002 (*Video<sub>1</sub>*), and R. Federer and M. Philippoussis at Wimbledon 2003 (*Video<sub>2</sub>*). As ground truth, the *Game Shot* containing net game segment is labeled with *Net game Shot* (NS), otherwise it is labeled with *Baseline game Shot* (BS), for the two players respectively. The detail information of the selected video data is listed in Table 1.

**Table 1.** Information of the experimental video data

Video	#Shot	# <i>Game Shot</i>	Top Half		Bottom Half	
			#NS	#BS	#NS	#BS
<i>Video<sub>1</sub></i>	881	316	58	258	230	86
<i>Video<sub>2</sub></i>	624	271	100	171	114	157
Sigma	1505	587	158	429	344	243

Half of the experimental data are selected randomly as training data set, and each *Game Shot* in training set is further divided and labeled with net-game segments and baseline-game segments, for top half and bottom half courts respectively. Subsequently, for certain half court, the temporal curves of the vertical and horizontal motion filters are segmented based on the labels, for training the two HMMs respectively. In recognition, all *Game Shot* with net-game segment detected are considered as NS, vice versa they are BS.

For comparison, the classification is performed based on the original MVs and the transformed MVs respectively. Table 2 illustrates the classification results based on the original MVs. When using the original MVs, the vertical motion responses between *Net Game* and *Baseline Game* can not be effectively distinguished as indicated in Fig. 5 (a). In the top half court, the *Net Games* have no salient vertical motion responses, and many of them are incorrectly classified into *Baseline Game*. In the bottom half court, lots of noise MVs cause some of the *Baseline games* having semblable vertical motion responses with *Net Games*, and thus many *Baseline Games* are misclassified into *Net Games*.

**Table 2.** Experimental results based on the original MVs

<i>Game Shot</i>	Top Half Court		Bottom Half Court	
	Precision(%)	Recall (%)	Precision(%)	Recall (%)
NS	30.91	37.78	64.55	67.78
BS	70.53	63.81	47.75	44.17

Table 3 illustrates the classification results based on the transformed MVs. With the transformed MVs, the performances are improved notably. As the vertical motion responses are greatly enlarged than that of the original MVs, the distinction between *Baseline Game* and *Net Game* are more salient, especially for the top half court as shown in Fig. 5 (b). The precision and recall rates of *Net Game* classification in top half court are both doubled.

**Table 3.** Experimental results based on the transformed MVs

<i>Game Shot</i>	Top Half Court		Bottom Half Court	
	Precision(%)	Recall (%)	Precision(%)	Recall (%)
NS	61.11	73.33	85.06	82.22
BS	87.50	80.00	74.60	78.33

Sometimes the block-based estimation algorithm of MVs has unavoidable mistakes and errors, under which the misclassification is unable to be corrected even by the MV transformation. Furthermore, as the players are small scale in proportion to the whole image, the noisy MVs may greatly disturb the player's

analysis. However, the experimental results indicate that in most conditions of tennis videos, the algorithm can properly revise the deformed MVs and enable the MV-based methods feasibly in the tennis video analysis.

## 5 Conclusion

An algorithm of motion vector transformation is proposed in this paper for the tennis video analysis. In this algorithm, the original deformed motion vectors are revised according to the player's true motion. Through such a transformation, it is more feasible to employ the MV-based methods in tennis video analysis. Experiments on classification of player's basic actions show very promising results. The future works may include: (i) make some improvements in setting up the transformation, such as the location of the tennis court lines, (ii) reduce the disturbance of random noises caused by MV estimation, and (iii) try more applications in tennis analysis based on the MV transformation.

## References

1. L.Y. Duan, M. Xu, T.S. Chua, Q. Tian, and C.S. Xu, "A Mid-level Representation Framework for Semantic Sports Video Analysis", *Proc. of the 11th ACM International Conference on Multimedia*, pp. 33–44, Berkeley, CA, USA, Nov. 2–8, 2003.
2. Y.F. Ma and H.J. Zhang, "A New Perceived Motion based Shot Content Representation", *Proc. of IEEE International Conference on Image Processing*, Thessaloniki, Greece, Vol. 3, pp. 426–429, Oct. 7–10, 2001.
3. G. Xu, Y.F. Ma, H.J. Zhang, and S.Q. Yang, "Motion based Event Recognition Using HMM", *Proc. of the 16th International Conference on Pattern Recognition*, Quebec, Canada, Vol. 2, pp. 831–834, Aug. 11–15, 2002.
4. G. Sudhir, John C.M. Lee, and Anil K. Jain, "Automatic Classification of Tennis Video for High-Level Content-Based Retrieval", *Proc. of 1998 International Workshop on Content-Based Access of Image and Video Databases*, pp. 81–90, Bombay, India, Jan. 03–03, 1998.
5. G.S. Pingali, Y. Jean, and I. Carlbom, "Real Time Tracking for Enhanced Tennis Broadcasts", *Proc. of 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 260–265, Santa Barbara, CA, USA, Jun. 23–25, 1998.
6. H. Miyamori, "Improving Accuracy in Behaviour Identification for Content-based Retrieval by Using Audio and Video Information", *Proc. of the 16th International Conference on Pattern Recognition*, Vol. 2, pp. 826–830, Quebec, Canada, Aug. 11–15, 2002.
7. H.J. Di, L. Wang, and G.Y. Xu, "A Three-step Technique of Robust Line Detection with Modified Hough Transform", *Proc. of SPIE Symposium on Multispectral Image Processing and Pattern Recognition*, pp. 835–838, Beijing, China, Oct. 20–22, 2003.
8. <http://www.hickoksports.com/glossary/gtennis.shtml>
9. L.R. Rabiner and B.H. Juang, "An Introduction to Hidden Markov Models", *IEEE Acoustics, Speech and Signal Processing Magazine*, vol. 3, pp. 4–16, Jan., 1986.

# Semantic Event Detection in Sports Through Motion Understanding<sup>\*</sup>

N. Rea<sup>1</sup>, R. Dahyot<sup>1,2</sup>, and A. Kokaram<sup>1</sup>

<sup>1</sup> Electronic and Electrical Engineering Department,  
University of Dublin, Trinity College Dublin, Ireland.

<sup>2</sup> University of Cambridge, Trumpington Street,  
Cambridge CB2 1PZ, United Kingdom.  
`oriabhan@tcd.ie`

**Abstract.** In this paper we investigate the retrieval of semantic events that occur in broadcast sports footage. We do so by considering the spatio-temporal behaviour of an object in the footage as being the embodiment of a particular semantic event. Broadcast snooker footage is used as an example of the sports footage for the purpose of this research. The system parses the sports video using the geometry of the content in view and classifies the footage as a particular view type. A colour based particle filter is then employed to robustly track the snooker balls, in the appropriate view, to evoke the semantics of the event. Over the duration of a player shot, the position of the white ball on the snooker table is used to model the high level semantic structure occurring in the footage. Upon collision of the white ball with another coloured ball, a separate track is instantiated allowing for the detection of pots and fouls, providing additional clues to the event in progress.

## 1 Introduction

Research interests in high-level content based analysis, retrieval and summarisation of video have grown in recent years [1]. A good deal of the interest has been focused on the detection of semantic events that occur in sports video footage [2,3]. This has been fueled primarily by the commercial value of certain sports and by the demands of broadcasters for a means of speeding up, simplifying and reducing the costs of the annotation processes. Current techniques used for annotating sports video typically involve loggers manually accounting for the events taking place [1]. The existing manually derived metadata can be augmented by way of automatically derived low level content-based features such as colour, shape, motion and texture [4]. This enables queries against visual content as well as textual searches against the predefined annotations allowing for more subjective queries to be posed.

---

<sup>\*</sup> Work sponsored by Enterprise Ireland Project MUSE-DTV (Machine Understanding of Sports Events for Digital Television), CASMS (Content Aware Sports Media Streaming) and EU-funded project MOUMIR (MOdels for Unified Multimedia Information Retrieval).

As humans operate at high levels of abstraction and since the most natural means for the lay person to query a corpus of data is through the use of semantics, it makes sense to develop algorithms that understand the nature of the data in this way. In order to do so, it becomes necessary to restrict the algorithms to a unique domain. These constraints enable low-level content based features to be mapped to high-level semantics through the application of certain domain rules.

The necessity for automatic summary generation methods for sports is highlighted by the fact that the semantic value of the footage spans short durations at irregular intervals. The remainder of the footage is generally of no consequence to the archetypal viewer (i.e. views of the crowd, breaks in play). Interesting events occur intermittently, so it makes sense to parse the footage at an event level. This offers the prospect of creating meaningful summaries while eliminating superfluous activities.

A common approach used to infer semantic events in sports footage is accomplished by modeling the temporal interleaving of camera views [5]. This is typically carried out using probabilistic modeling techniques such as HMMs or NNs. This inherent temporal structure of some broadcast sports is not however, evident in snooker footage. Thus, a model based on evolving camera views can not be used for the purposes of this research. Other works use deterministic methods [6], but are limited in some regards with respect to the adaptivity of the models to changes in playing conditions. In this paper, we propose a novel approach for the detection of semantic events in sports whereby the spatio-temporal behaviour of an object is considered to be the embodiment of a semantic event. For the case of snooker, in the appropriate camera view, the white ball is tracked using a colour based particle filter [7]. Parzen windows are used to estimate the colour distribution of the ball as it is a small object relative to the rest of the image. The implementation of the particle filter allows for ball collision detection and ball pot detection. A separate ball track is instantiated upon detection of a collision and the state of the new ball can be monitored. Detection of such events augment the HMM decision process by providing a binary classifier where uncertainty is present. The evolution of the white ball position is modeled using a discrete HMM. Models are trained using six subjective human perceptions of the events in terms of their perception of the evolving position of the white ball. The footage is parsed and the important events are automatically retrieved.

## 2 Shot Classification

Similar to other sports, the finite number of fixed camera views used in broadcast sports footage are arranged in such a way as to cause the viewer to become immersed in the game while trying to convey the excitement of the match to a mass audience. In snooker, the typical views used are those of the full-table, close-ups of the player or crowd, close-ups of the table and assorted views of the table from different angles.

For the purpose of this research we consider the most important view to be that of the full table. Analysis on 30 minutes of televised footage from three

different broadcast sources shows it to occupy approximately 60% of the total coverage duration. In this view all balls and pockets on the table are visible, enabling ball tracking and pot detection. It is therefore necessary to ensure that the camera views can be classified with high precision.

Shot classification is accomplished using the method outlined in [8]. The footage is parsed at a clip level based on the geometrical content of the camera views. This approach does not require extraction of 3D scene geometry and is generic to broadcast sports footage which exhibit strong geometrical properties in terms of their playing areas. The temporal evolution of the derived feature is modeled using a first-order discrete HMM, allowing the views to be correctly classified. The system for parsing snooker footage is illustrated in figure 1. The relevant full table shots are passed to an event processor where tracking, pot detection and foul detection are performed.

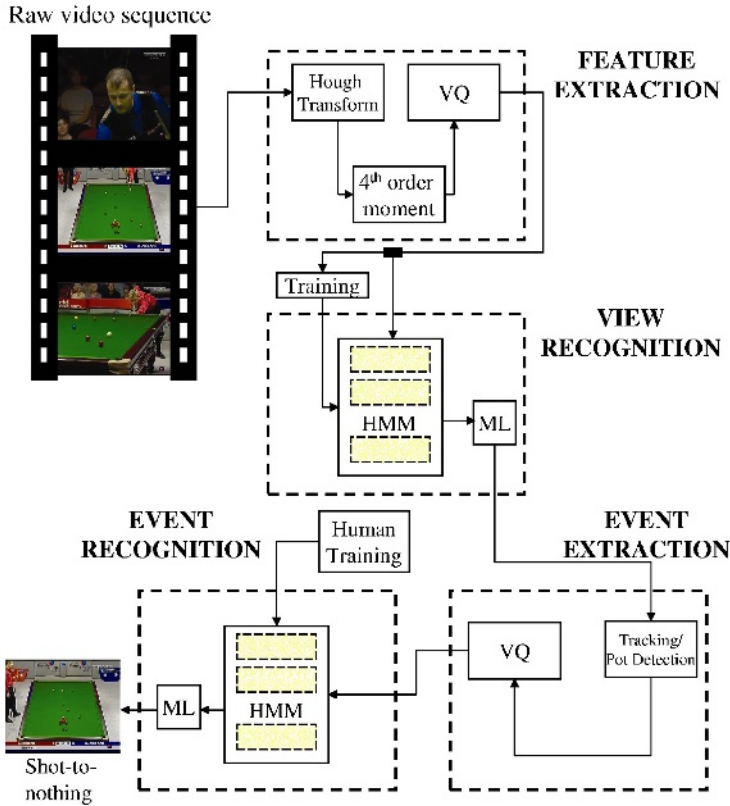


Fig. 1. System for parsing broadcast snooker footage.

### 3 Event Classification

It was observed that the track drawn out by the white ball over the duration of a player's shot characterises an important event. If the spatio-temporal evolution of the white ball's position can be modeled, semantic episodes in the footage can be classified. We must firstly define the events of interest that occur in snooker in terms of the spatio-temporal evolution of the position of the white ball and how pots and fouls affect the event semantics.

#### 3.1 Events of Interest in Snooker and Game Heuristics

In snooker, players compete to accumulate the highest score possible by hitting the white ball and potting the coloured balls in a particular sequence. The coloured balls vary in value from one (red) to seven (black), so different strategies must be employed to gain and maintain control of the table. The occurrence of a ball pot or foul (the white ball not colliding with a coloured ball at all) will affect the viewer's perception of the event in hand. Priori domain knowledge makes use of these events, allowing a set of heuristics to be established which are used to evaluate the current maximum likelihood classification upon detection of a foul or a pot. This is illustrated in figure 3.

The 'plays' we consider are characterised by the spatio-temporal behaviour of the white ball as follows (where  $C$  is the event number) and are affected by the state of the coloured ball (pot/no pot) and the white ball (foul/no foul).

**Break-building:**  $C = 1$ . As the player attempts to increase his score he will try and keep the white ball in the center of the table with easy access to the reds and high valued balls. If a pot has been detected, the player is attempting to build a high break ( $C = 1$ ) (figure 2). In the unlikely event of one of the balls not being potted, the white ball will probably be in a position such that the remaining balls will be eminently 'potable'. This is called an 'open table' event ( $C = 5$ ).

**Conservative play:**  $C = 2$ . Similar to the shot-to-nothing, except a coloured ball will not be potted when the white navigates the full length of the table. If this model is chosen as being the most likely, and a pot is detected, a shot-to-nothing will be inferred ( $C = 4$ ). This is because the ball will be in an area where it might prove difficult for a player to pot the next coloured ball in the sequence.

**Escaping a snooker:**  $C = 3$ . If the player is snookered (no direct line of sight to a ball) he will attempt to nestle the white amongst the reds or send the white ball back to top of the table. If a pot is detected following the classification of a snooker escape, the heuristics will infer a break-building event ( $C = 1$ ). As the only goal of the player will be to escape the snooker without conceding a foul or an open table if a ball is potted, it simply serves as a bonus.

**Shot-to-nothing:**  $C = 4$ . The white ball is hit from the top of the table, traverses the table, and returns back to the top of the table. If a pot is

detected, the pot heuristics will infer a shot-to-nothing ( $C = 4$ ) (figure 2). If there is no pot, the spatio-temporal evolution of the white ball position will show that the player is attempting to return the white ball to the top of the table. A conservative play event, ( $C = 2$ ), could therefore be inferred as he is making the next shot as difficult as possible for his opponent.

In all of these cases a foul by the white, flagged by a non-instantiated second track, or if the white is potted will result in a foul ( $C = 6$ ) being inferred. Play will then be transferred to the opposing player.

It was also observed that a snooker escape event is characterised by a cut from the full-table view to a close up view of the ball about to be hit. This occurs while the white ball is still in motion. If the velocity of the white ball,  $V > 0$ , a snooker escape is inferred (figure 3).

### 3.2 Motion Extraction

The proposed approach is similar to those methods used in handwriting recognition [9]. The position of the input device in these systems is easily obtainable through a stylus/pad interface. In the case of snooker however, the exact position of the white ball is not so readily available. Having located the full table views in the footage [8], a robust colour based particle filter is employed in order to keep track of the position of the white ball in each frame and simultaneously track the first ball hit.

**Localisation of the white ball:** Events within clips are found by monitoring the motion of the white ball. As there is no camera motion in the full table view, the white is initially located by finding the brightest moving object on the table as it first starts moving. The semantic episode begins when the white ball starts moving and ends when it comes to rest. The implementation of the particle filter trivialises the accretion of these velocity values.

### 3.3 Ball Tracking

The tracker used in this work is similar to that implemented in [7]. The objects to be tracked however are significantly smaller (approximately 100 pels in size). We use the HSV colour space for our colour based probabilistic tracker. In order to facilitate an increase in resolution by selecting a small object relative to the size of the image, the colour distribution needs to be extended for both target and candidate models. Parzen windows are used to estimate the distribution of the hue and saturation components while the luminance component is quantised to 16 bins to reduce the effect of the lighting gradient on the table.

A target model of the ball's colour distribution is created in the first frame of the clip. Advancing one frame, a set of particles is diffused around the projected ball location using a deterministic second order auto-regressive model and a stochastic Gaussian component. Histograms of regions the same size as the ball are computed using the particle positions as their centers. A





**Fig. 2.** Tracking and table sections. Left to right: Shot-to-nothing; Break building; Spatial segmentation of the table.

Bhattacharyya distance measure is used to calculate the similarity between the candidates and the target which is in turn used to weight the sample set,  $X = \left\{ \left( x_k^{(n)}, w_k^{(n)} \right) \mid n = 1 \dots N \right\}$ , where  $N$  is the number of particles used. The likelihood of each particle is computed:

$$w_k^{(n)} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\left(1 - \sum_{j=1}^m \sqrt{\rho(x_k^{(n)})^j \xi^j}\right)}{2\sigma^2}} \quad (1)$$

$\rho(x_k^{(n)})$  is the histogram of the candidate region at position  $x_k$  for sample  $n$ ,  $\xi$  is the target histogram and  $m$  is the number of histogram bins and  $\sigma^2 = 0.1$ .

### 3.4 Collision Detection

A ball collision is detected by identifying changes in the the ratio between the current white ball velocity  $v_k$  and the average previous velocity  $v_p$  (defined below, where  $d$  is the frame where the white starts its motion).

$$\mathbf{v}_p = \frac{1}{(k-2)-d} \left( \sum_{i=d}^{k-2} \mathbf{v}_i \right) \quad (2)$$

If the ball is in the vicinity of the cushion, a cushion bounce is inferred and  $d$  is set to the current frame. Ratios in the x and y velocity components  $v_k^x/v_p^x, v_k^y/v_p^y$  are analysed to isolate changes in different directions. A collision is inferred when the condition in equation 3 is satisfied.

$$h_k = \left\{ \left( \frac{|\mathbf{v}_k^x|}{|\mathbf{v}_p^x|} < 0.5 \right) \wedge \left( \frac{|\mathbf{v}_k^y|}{|\mathbf{v}_p^y|} > 0.5 \right) \right\} \vee \left\{ \left( \frac{|\mathbf{v}_k^y|}{|\mathbf{v}_p^y|} < 0.5 \right) \wedge \left( \frac{|\mathbf{v}_k^x|}{|\mathbf{v}_p^x|} > 0.5 \right) \right\} \vee \left\{ \left( \frac{|\mathbf{v}_k^x|}{|\mathbf{v}_p^x|} < 0.5 \right) \wedge \left( \frac{|\mathbf{v}_k^y|}{|\mathbf{v}_p^y|} < 0.5 \right) \right\} \quad (3)$$

The condition therefore flags an event when velocity changes by 50%. The form of the decision arises because the physics of colliding bodies implies that at collision, changes in velocity in one direction are typically larger than another except in the case of a ‘flush’ collision where a reduction of  $< 50\%$  in both directions is exhibited.

**Pot detection:** Distinguishing between correct tracking and the loss of a ‘lock’ can be accomplished by using a threshold on the sum of the sample likelihoods,  $L_l$ . If the cumulative likelihood at time  $k$ ,  $L^k > L_l$  a correct lock is assumed, and the ball has been found. If  $L^k/L^{k-1} < 0.5$ , the ball has been potted.

### 3.5 Spatial Encoding of the Table

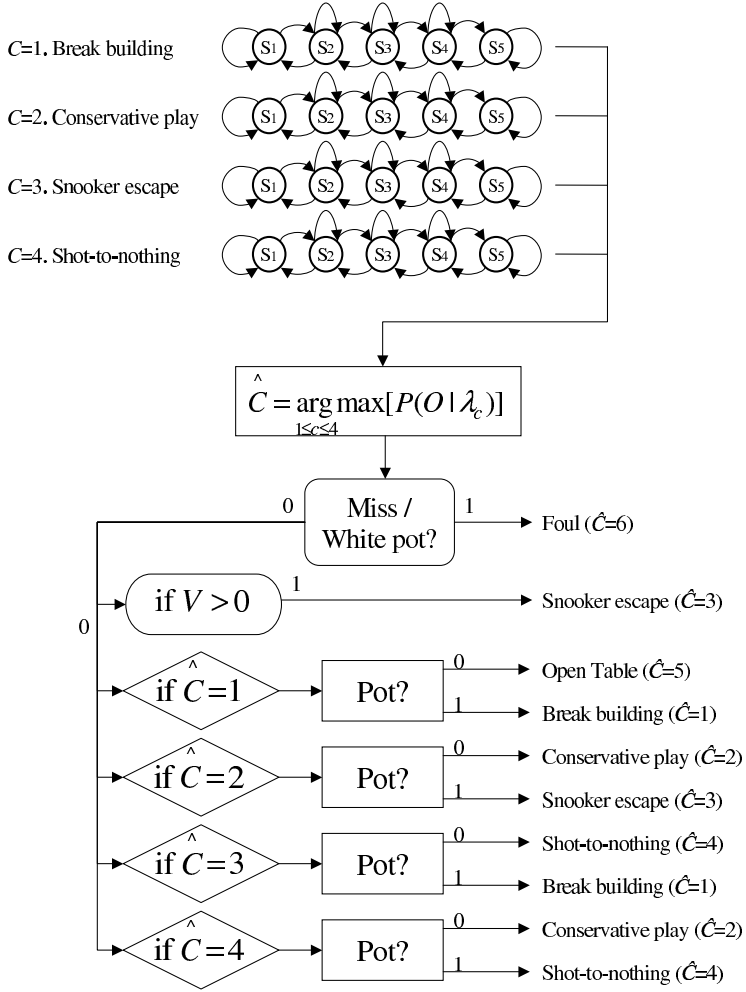
The dimensions of the table, the positions of the balls and their values dictate the flow of the play to be mostly along the long side of the table (or without loss of generality, along the vertical). The temporal behaviour of the vertical position of the white alone could therefore be considered to embody a semantic event. Using the fact that diagonals of a trapezoid intersect at its center, the table can be divided into 5 sections at the coloured ball’s spot intervals (figure 2). Initially, the table is divided by intersecting the main diagonals, retrieving the center line. Sub division of the two resulting sections retrieves the pink and brown lines, and so on. The starting and end positions of the white ball alone do not sufficiently represent a semantic event. The model must be augmented by the dynamic behaviour of the white ball. The observation sequence,  $O$ , is the sequence of evolving table sections.

### 3.6 Establishing the Model Topology

Modeling the temporal behaviour of the white ball in snooker is accomplished using a first order HMM. HMMs have been shown to be one of the most efficient tools for processing dynamic time-varying patterns and allow a rich variety of temporal behaviours to be modeled. The model topology is derived from the observations, reflecting the nature of the target patterns. A left-to-right/right-to-left topology is chosen to model the motion of the white ball for each event, revealing the structure of the events in state form. Each section is represented by a state in the HMM where the state self-transitions introduce time invariance as the ball may spend more than one time-step in any one section.

Knowing the number of states (or sections of the table),  $N = 5$ , and discrete codebook entries,  $M = 5$ , a model  $\lambda$ , can be defined for each of the competing events. A succinct definition of a HMM is given by  $\lambda_c = (A_c, B_c, \pi_c)$ , where  $c$  is event label. The model parameters are defined as:  $A$ , the state transition probability matrix,  $B$ , the observation probability matrix, and  $\pi$  a vector of initial state probabilities.

The Baum-Welch algorithm is used to find the maximum likelihood model parameters that best fit the training data. As the semantic events are well understood in terms of the geometrical layout of the table, the models can be trained using human understanding. Six separate human perceptions of the events listed in section 3.1 were formed in terms of the temporally evolving table coding sequence of the white ball. The models used are shown in figure 3 with an example of a single training sequence. The models are initialised by setting  $\pi^n = 1$  where  $n = O_1$ .



**Fig. 3.** Event HMMs with pot and foul classifiers.

Each semantic episode can then be classified by finding the model that results in the greatest likelihood of occurring according to equation 4.

$$\hat{C} = \arg \max_{1 \leq c \leq C} [P(O|\lambda_c)], \quad C = 4 \text{ events.} \quad (4)$$

## 4 Results

Experiments were conducted on two footage sources ( $F1, F2$ ) from different broadcasters of 17.5 and 23.2 minutes in duration. 21 occurrences of the events

to be classified were recognised in  $F1$ , of which 11 were break-building, 6 were conservative plays, 2 shot-to-nothings, 1 open-table, 0 snooker escapes and 1 foul. 30 events occurred in  $F2$  of which there were 16 break-building, 8 conservative plays, 1 shot-to-nothing, 2 open tables, 2 snooker escapes and 1 foul. The classification results are assessed by computing the recall (R) and the precision (P).

$$R = \frac{A}{A+C} \quad P = \frac{A}{A+B} \quad (5)$$

$A$  is the number of correctly retrieved events,  $B$  the number of falsely retrieved events and  $C$  the number of missed events.

**Table 1.** Event classification results.

Event type	$F1$ (P)	$F1$ (R)	$F2$ (P)	$F2$ (R)
Break-building ( $C = 1$ )	91.67%	100%	94.12%	100%
Conservative play ( $C = 2$ )	100%	100%	100%	75%
Snooker escape ( $C = 3$ )	N/A	N/A	100%	100%
Shot-to-nothing ( $C = 4$ )	100%	50%	100%	100%
Open Table ( $C = 5$ )	100%	100%	66%	100%
Foul ( $C = 6$ )	100%	100%	50%	100%

In  $F1$  the only misclassification was that of a shot-to-nothing being classified as a break building event. In  $F2$  a problem arose in the classification of two conservative plays. One was misclassified as a foul due to light contact being made by the white with a coloured ball and a collision was not detected, while the second was misclassified as an open table event.

## 5 Discussion

In this paper we have considered the dynamic behaviour of an object in a sport as being the embodiment of semantic episodes in the game. Modeling the temporal evolution of the low level feature in this way allows important episodes to be automatically extricated from the footage. Results obtained are promising using the most relevant 60% of footage. Augmenting the feature set with more tracking information could improve the retrieval further. We are currently attempting to use the same process to classify semantic episodes that occur in broadcast tennis footage. Furthermore, we are investigating the possibility of generating game summaries where the excitement of each match could be gauged by the frequency of different events.

## References

1. Bertini, M., Bimbo, A.D., Nunziati, W.: Semantic annotation for live and posterity logging of video documents. In: Visual Communications and Image Processing (VCIP 2003). (2003)
2. Kijak, E., Gros, P., Oisel, L.: Temporal structure analysis of broadcast tennis video using hidden markov models. In: SPIE Storage and Retrieval for Media Databases. (2003) 289–299
3. Assfalg, J., Bertini, M., Bimbo, A.D., Nunziati, W., Pala, P.: Soccer highlight detection and recognition using hmms. In: IEEE International Conference on Multimedia and Expo. (2002)
4. Djeraba, C.: Content-based multimedia indexing and retrieval. *IEEE Multimedia* 9 (2002) 52–60
5. Chang, P., Han, M., Gong, Y.: Extract highlights from baseball game video with hidden markov models. In: Proceedings of the International Conference on Image Processing (ICIP '02). (2002)
6. Ekin, A., Tekalp, A.M., Mehrotra, R.: Automatic soccer video analysis and summarization. In: International Conference on Electronic Imaging: Storage and Retrieval for Media Databases. (2003) 339–350
7. Perez, P., Hue, C., Vermaak, J., Gangnet, M.: Colour based probabilistic tracking. In: European Conference on Computer Vision 2002 (ECCV 2002). (2002)
8. Denman, H., Rea, N., Kokaram, A.C.: Content based analysis for video from snooker broadcasts. *Journal of Computer Vision and Image Understanding (CVIU): Special Issue on Video Retrieval and Summarization* 92 (2003) 141–306
9. Lee, J.J., Kim, J., Kim, J.H.: Data-driven design of hmm topology for on-line handwriting recognition. In: The 7th International Workshop on Frontiers in Handwriting Recognition. (2000)

# Structuring Soccer Video Based on Audio Classification and Segmentation Using Hidden Markov Model

Jianyun Chen, Yunhao Li, Songyang Lao, Lingda Wu, and Liang Bai

Multimedia Research & Development Center,  
National University of Defense and Technology, ChangSha 410073, P. R. China  
cjy2918@163.com

**Abstract.** This paper presents a novel scheme for indexing and segmentation of video by analyzing the audio track using Hidden Markov Model. This analysis is then applied to structuring the soccer video. Based on the attributes of soccer video, we define three audio classes in soccer video, namely *Game-audio*, *Advertisement-audio* and *Studio-audio*. For each audio class, a HMM is built using the clip-based 26-coefficients feature stream as observation symbol. The Maximum Likelihood method is then applied for classifying test data using the trained models. Meanwhile, considering that it is highly impossible to change the audio types too suddenly, we apply smoothing rules in final segmentation of an audio sequence. Experimental results indicate that our framework can produce satisfactory results.

## 1 Introduction

Studies have been reported in the literature addressing sports video structuring. Prior works include syntactical segmentation [1, 2] and semantic annotation [3, 4]. And people pay much effort to the image sequence. But video sequence is a rich multimodal information source, containing audio, text, image, etc. Efficient indexing and retrieval of video requires taking multi-cues from video sequence into account. Audio as a counterpart of visual information in video sequence got more attention recently for video content analysis [5, 6]. On the other hand, Hidden Markov Model [7] has good capability to grasp the temporal statistical property of stochastic process and to bridge the gap between the low-level features from video data and the high-level semantics the users are interested in. The emphasis of this paper is applying the HMM to automatic audio classification and segmentation for final soccer video structuring. The clip-based audio features are extracted and used to train the HMMs of the three kinds of soundtrack in soccer video. The smoothing rules improve classification and segmentation accuracy. The results and related discussions are given.

The rest of this paper is organized as follows. In Section 2, we summarize three audio classes in soccer video and propose the framework of automatic audio classification and segmentation for soccer video structuring. In Section 3, we explain the

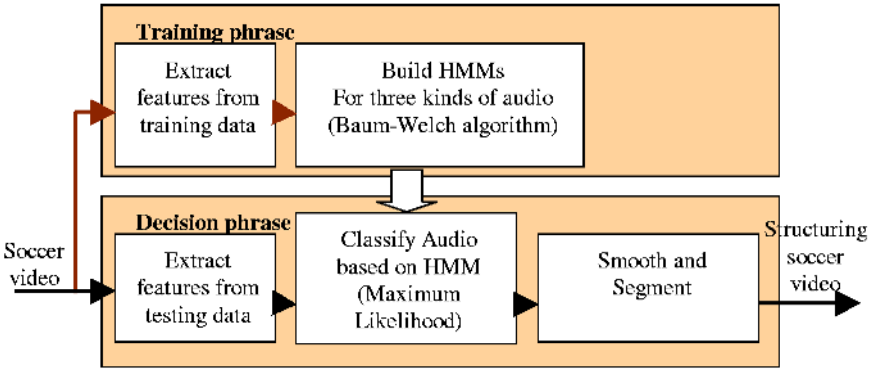


**Fig. 1.** Typical sequence of segments in soccer video

modules of our system: audio features extraction, HMM training, testing data classification, smoothing and segmentation. Section 4 is experimental results and discussion. At last the conclusions and future works are given in Section 5.

## 2 The Framework of Audio Classification and Segmentation for Structuring Soccer Video

Figure 1 contains a simplified diagram of a typical sequence of segments in soccer video. The whole soccer video is composed of five sequential semantic parts: First-half game, Advertisement, Studio, Advertisement and Second-half game. Usually, the soundtrack of Game is noisy speech; the one of Advertisement is the mixture of speech and music; while the one of Studio is pure speech. These three kinds of soundtracks that correspond to different segments have distinct low-level features. That is to say, video structuring is achieved as long as the soundtrack is classified and segmented. Therefore, we define three types of audio in soccer video for automatic audio classification and segmentation, namely *Game-audio*, *Advertisement-audio* and *Studio-audio*.



**Fig. 2.** The framework of automatic audio classification and segmentation in soccer video based on HMMs

Then we can convert the video indexing problem into the audio classification problem. Unlike previous approaches, we want to propose a stochastic model rather than fine-tuning one. We want to expand on these three kinds of audio for many more

audio-events. With this in mind, we use the Hidden Markov Model for automatic audio classification and segmentation in soccer video.

Diagram of our system is shown in Figure 2. Automatic audio classification and segmentation for soccer video structuring is processed in two steps: the first is training phrase and the second is decision phrase. In training phrase, clip-based audio features are extracted from training data and three HMMs are built using Baum-Welch algorithm. Then, in decision phrase, the same audio features are got from testing data. Classify the audio sequence with the Maximum Likelihood method based on the three HMMs from the first step. Finally the smoothing rules are used to improve the segmentation accuracy. Thus the final soccer video structuring is finished. We explain the modules of two phrases in detail in next section.

### 3 Audio Classification and Segmentation for Structuring Soccer Video Using HMMs

HMM has been successfully applied in several large-scale laboratory and commercial speech recognition systems. In traditional speech recognition system, a distinct HMM is trained for each word or phoneme, and the observation vector is computed every frame (10-30ms). Here we do not need to grasp the detail information at the resolution of several milliseconds. What we are interested in is the semantic content that can only be determined over a longer duration. Based on this observation, the basic classification unit in our system is not a frame, but a clip.

#### 3.1 Audio Features Extraction [8]

The audio signal is sampled at 22050 Hz and 16 bits/sample. The audio stream is then segmented into clips that are 1 second long with 0.5 second overlapping with the previous ones. Each clip is then divided into frames that are 512 samples long. For each frame, we extract the feature vector with 26 coefficients as follows: the sound-track is preprocessed using a 40 channels filter-bank constructed using 13 linearly-spaced filters followed by 27 log-spaced filters. Cepstral transformation gives 12 mel frequency cepstral coefficients (MFCC), 12 MFCC Delta and 2 energy coefficients. The MFCC Delta is computed as

$$\Delta c_n(m) = \sum_{k=-2}^2 k c_{n-k}(m) * 0.56, 1 \leq m \leq P. \quad (1)$$

Where  $c$  stands for MFCC coefficient,  $\Delta c$  is MFCC Delta coefficient and  $P$  is defined as 12.

So for each clip, this gives a 26 coefficients feature streams.



### 3.2 Building HMMs for Three Audio Classes

We borrow the Hidden Markov Models (HMM) from the speech recognition field for automatic audio classification and segmentation, where they have been applied with great success [7,8].

We model three audio classes using a set of states with a Markovian state transition and a Gaussian mixture model for observation probability density in each state.

We use continuous density model in which each observation probability distribution is represented by a mixture density. For state  $j$ , the probability  $b_j(O_t)$  of generating observation  $O_t$  is given as follow:

$$b_j(O_t) = \sum_{m=1}^{M_j} c_{jm} G(\mu_{jm}, \Sigma_{jm}, O_t). \quad (2)$$

Where  $G(\mu, \Sigma, O)$  is the multivariate Gaussian function with mean  $\mu$  and covariance matrix  $\Sigma$ ,  $M_j$  is the number of mixture components in state  $j$  and  $c_{jm}$  is the weight of the  $m$ th component. In our system the observation symbol is the 26 coefficients vector as we mentioned earlier, and  $M_j$  is defined as 15.

For each audio class, an ergodic Hidden Markov Model with 3 states is used to model the temporal characteristics. With  $q_t$ , denoting the state at instant  $t$  and  $q_{t+1}$  the state at  $t+1$ , elements of matrix  $A$  are given as follows:

$$a_{ij} = P(q_{t+1} = j \mid q_t = i). \quad (3)$$

The parameters of the model to be learnt are the state transition probability distribution  $A$ , the observation symbol probability distribution  $B$  and the initial state distribution  $\pi$ . The model is simply referred to as  $\lambda = (A, B, \pi)$ . The Baum-Welch re-estimation procedure is used to train the model and learn parameters  $\lambda$ .

In theory, the re-estimation procedure should give values of the HMM parameters which correspond to a local maximum of the likelihood function. Therefore a key question is how we choose initial estimates of the HMM parameters so that the local maximum is the global maximum of the likelihood function. Here, initial estimates are obtained by segmental  $k$ -means procedure.

To solve the problem of underflow in training, we perform the computation by incorporating a scaling procedure. Here, in order to have sufficient data to make reliable estimates of all model parameters, we use multiple observation sequences. The modification of the re-estimation procedure is straightforward and goes as follows. We denote the set of  $K$  observation sequences as  $O = [O^{(1)}, O^{(2)}, \dots, O^{(K)}]$ , where  $O^{(K)}$  is the  $k$ th observation sequence. Then:

$$\bar{\pi}_j = \frac{1}{K} \sum_{k=1}^K \gamma_1^{(k)}(j). \quad (4)$$

$$\bar{a}_{ij} = \sum_{k=1}^K \sum_{t=1}^{T_k-1} \xi_t^{(k)}(i, j) / \sum_{k=1}^K \sum_{t=1}^{T_k-1} \gamma_t^{(k)}(i). \quad (5)$$

$$\bar{b}_j(k) = \sum_{l=1}^K \sum_{t=1}^{T_l} \gamma_t^{(l)}(j) / \sum_{l=1}^K \sum_{t=1}^{T_l} \gamma_t^{(l)}(j). \quad (6)$$

$s.t. o_t^{(l)} = v_k$

Where  $\gamma_t(j)$  denotes the probability of being in state  $j$  at time  $t$ , and  $\xi_t(i, j)$  denotes the probability of being in state  $i$  at time  $t$  while state  $j$  at time  $t+1$ .

### 3.3 Audio Classification and Segmentation Using HMMs

Once the parameters are learnt with the training data, the models can then be used to perform maximum likelihood classification for each clip. The classification approach leads to segmentation of the audio stream where each segment gets the label of the classified model. This label can be used along with the temporal information for indexing. The likelihood assigned by the classification to each label reflects a degree of confidence in the accuracy of the label. This can be used to avoid hard threshold while indexing.

Thus segmentation of an audio stream is achieved by classifying each clip into an audio class in soccer video. Meanwhile, considering that the audio stream is always continuous in video program, it is highly impossible to change the audio types too suddenly or too frequently. Under this assumption, we apply smoothing rules in final segmentation of an audio sequence [6]. The smoothing rule is:

$$\text{Rule if } (c[1] \neq c[0] \&\& c[2] = c[0]) \text{ then } c[1] = c[0]. \quad (7)$$

Where three consecutive clips are considered,  $c[1], c[0], c[2]$  stand for the audio class of current clip, previous one and next one respectively. This rule implies that if the middle clip is different from the other two while the other two are the same, the middle one is considered as misclassification.

After smoothing the classification results, the segmentation accuracy is improved and the final segmentation is finished.

## 4 Experimental Results and Discussion

Three soccer video programs used in our experiment are briefly described in Table 1.

**Table 1.** Soccer video programs used in our experiment

No.	Soccer video Name	Length	Source
Soccer1	English premier football league: Astonvilla vs. Manchester utd	24m38s	Sports channel of HNTV on 03/15/2003
Soccer2	English premier football league: Fulham vs. Manchester utd	25m2s	Sports channel of HNTV on 03/22/2003
Soccer3	German Division One League: Bayern Munich vs. Leverkusen	10m53s	Sports channel of HNTV on 09/20/2003

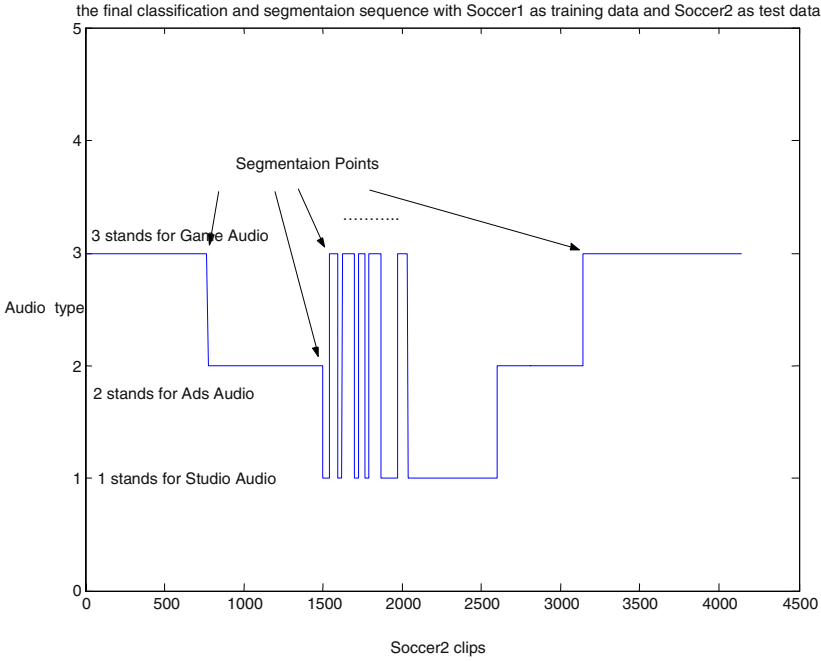
In our experiments, HMMs are trained on one program and tested on other programs. This process is repeated three times [2]. The first result measurement is the *classification accuracy*, defined as the number of correctly classified clips over total number of clips. Training and testing accuracies are shown in Table 2. *Average classification accuracy (avg-cla)* of each program as testing data is computed as the mean of elements of current row; similarly, *average generalization accuracy (avg-gen)* is computed for the program as training data; and the overall *average classification/generalization accuracy* over the entire dataset is put in the lower right corner. From Table 2, it is easily found that the algorithm performance is satisfactory.

**Table 2.** Classification accuracy

Testing Data		Training Data			Avg-cla
		Soccer1	Soccer2	Soccer3	
Soccer1	<i>Game-audio</i>	0.9196	0.8228	0.7998	0.8474
	<i>Ads-audio</i>	0.9310	0.8598	0.8964	0.8957
	<i>Studio-audio</i>	0.8646	0.8917	0.88	0.8788
Soccer2	<i>Game-audio</i>	0.8420	0.9891	0.8758	0.9023
	<i>Ads-audio</i>	0.9400	0.9626	0.8598	0.9208
	<i>Studio-audio</i>	0.8486	0.7810	0.8333	0.8210
Soccer3	<i>Game-audio</i>	0.8793	0.8470	0.9347	0.8870
	<i>Ads-audio</i>	0.8816	0.9075	0.9208	0.9033
	<i>Studio-audio</i>	0.8114	0.8767	0.9066	0.8649
<i>Avg-gen</i>		0.8798	0.8820	0.8756	0.88

Since our goal is to do joint classification and segmentation in one-pass, we are also interested in measuring the segmentation results. The final classification and segmentation sequence with Soccer2 as testing data and Soccer1 as training data is shown in Figure3. For Soccer2, the program begins with Game, followed by the commercial breaks in half time. Then the video is continued with Studio scene. It is noticeable that between the 1500<sup>th</sup> clip and the 2000<sup>th</sup> clip the Studio audio is frequently interleaved with Game audio. This outcome accord with the fact: in the Stu-

dio scene of Soccer2, when the anchorman comments on the first-half game, the program is usually switched to the corresponding game scene. Then the following sequence is another period of time of advertisement and second-half game. Obviously, we take the step points as the segmentation points.



**Fig. 3.** The final classification and segmentation sequence with Soccer2 as testing data and Soccer1 as training data

We define *segmentation-point-offset* be the absolute clip difference between the nearest segmentation point in detection result and every segmentation point in the ground-truth. And the distribution of *segmentation-point-offset* over all testing condition is used to measure the segmentation accuracy. The result shown in Table3 indicates that more than 70% of the segmentation points are detected within a 3-clips long window.

**Table 3.** *Segmentation-point-offset* Distribution

<i>Segmentation-point-offset</i>	[0,5)	[6,10)	[11,15)	[16,20)	$\geq 20$
Percentage	73%	5%	8%	6%	8%

## 5 Conclusions and Future Works

In this paper, we have described a novel soccer video structuring approach using audio features. We develop a Hidden Markov Model based on the characteristics of

audio in soccer video. The classes supported are *Game-audio*, *Advertisement-audio* and *Studio-audio*. Making use of these three audio classes, we index and segment the soccer video. The preliminary experiments indicate the proposed technique is feasible and promising. Our framework is generic enough to be applicable to other sports video, such as tennis, volleyball etc, and even other video type. And it can be applied to the detection of other audio-events because in our system there are not specific algorithm or threshold tune-ups.

In future works, we will enhance the performance of our work in two ways: Since HMM is the an efficient model of mapping the low-level observations and high-level semantics, the first direction for future research is to invent better features for better performance. On one hand, we should go deep into audio processing; on the other hand, combing with visual information ought to be noticed. Then another future work is the improvement of Hidden Markov Model. Unfortunately, there is no simple, theoretically correct, way of choosing the type of model (ergodic or left-right or some other form), choosing of model size (number of states) etc. So we can choose different numbers of states and different numbers of mixture components for the HMM to improve the accuracy. In this direction, the focus is automatically deciding these parameters using some optimality criteria.

## References

1. D. Zhong and S. F. Chang. Structure Analysis of Sports video Domain Models. In IEEE Conference on Multimedia and Expo, (2001) 920-923
2. L.Xie, S. F. Chang, A. Divakaran et al. Structure analysis of soccer video with Hidden Markov models. In Proc. ICASSP, Orlando, FL, (2002)
3. Yihong Gong, Lim Teck Sin, Chua Hock Chuan, et al. Automatic Parsing of TV Soccer Programs, In IEEE International Conference on Multimedia Computing and Systems, Washington D.C, (1995)
4. A. James and S. F. Chang. Automatic Selection of Visual Features and Classifiers, In SPIE Conference on Storage and Retrieval for Media Database, Vol.3972, San Jose CA, (2000) 346-358
5. Liu Zhu, Wang Y, et al. Audio Feature Extraction and Analysis for Scene Segmentation and Classification. Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology, 1/2, (1998), 61-79
6. Lie Lu, Hong-jiang Zhang and Hao Jiang. Content Analysis for Audio Classification and Segmentation. IEEE Transactions on Speech and Audio Processing, Vol.10, No.7, (2002)
7. Lawrence Rabiner, Biing-Hwang Juang. Theory and implementation of Hidden Markov Models, Book chapter, Fundamentals of speech recognition. Prentice Hall, (1993)
8. Yang XingJun, Speech Signal Processing, Publishing House of Electronics Industry, Beijing, PRC, (1995)

# EDU: A Model of Video Summarization

Yu-Xiang Xie, Xi-Dao Luan, Song-Yang Lao, Ling-Da Wu, Peng Xiao, and Jun Wen

Centre for multimedia Technology, National University of Defense Technology,  
Changsha, 410073, China  
{xyx89,xidaoluan}@sina.com, laosongyang@vip.sina.com

**Abstract.** A novel and expandable video summarization model called EDU is proposed. The model starts from video entities, gets utilities after descriptions, and finally generates video summarization according to the utilities. As a general expandable model, EDU has a description scheme to save the preprocessed information, and a utility function based on the descriptions. The concepts and structures of EDU model are described in detail, and a method of news story summarization based on this model is also proposed. The experiment proves the effectiveness of the method.

## 1 Introduction

With the rapid development of multimedia techniques, there emerge abundant digital videos, such as news, advertisements, surveillance videos, home videos, etc. These changes promote new techniques on the storage, indexing and accessing of videos. Among them, one important problem is how to browse large quantities of video data, and how to access and represent the content of videos. The technique of video summarization can solve these problems to some extent.

Video summarization is a short summary of the content of a long video document. It is a sequence of still or moving images representing the content of a video in such a way that the target party is rapidly provided with concise information about the content while the essential message of the original is well preserved [1].

Researches on the video summarization technique can be traced back to the Informedia project [2] developed by Carnegie Mellon University. Later, it was widely studied by various universities and organizations, such as Columbia University [3], AT&T laboratory, Intel Corporation, Mannheim University [5], Microsoft Research Asia [6], etc., and many advanced algorithms and methods have been proposed [7,8].

There are about six popular types of video summarizations, namely titles, posters, storyboards, skims and multimedia video summarizations. But most works fall short of a unified model to supervise the generation of video summarization. The purpose of this paper is to build a general video summarization model and realize news video summarization based on this model.

## 2 EDU Model

We propose a video summarization model- EDU, which is the abbreviation of Entity-Description-Utility. First, we'll introduce some concepts.

### 2.1 Related Concepts

**Definiton 1. Entity.** The so-called entity is the existence in videos. It can be notional, or physical. From top to bottom, we regard all video files, stories, scenes, shots and frames as entities. Entities at different levels form the structure of videos, and each entity has its attributes and predications. For example, frame is an entity, while each pixel is an attribute of the frame, and the position of each pixel is the predication. The high-level entity is formed from low-level entities; such as the shot entity is formed from many frame entities.

Entity is, in fact, a subset of videos. Supposing a video segment containing  $N$  shots, then the  $k$ th shot can be described in this way:

$$Shot_k = \{f_s \in P(f) \mid start(k) \leq f_s \leq end(k)\}, k \in [1, N] \quad (1)$$

Where  $f_s$  is a frame,  $P(f)$  is the set of all frames,  $start(k)$  and  $end(k)$  is the start and the end frame number of shot  $k$  respectively.

**Definition 2. Description.** Description is the abstract and general explanation of an entity. Different from the original information, description is the processed and extracted information, which can be more understandable. Different levels of entities have different descriptions. We define descriptions of entities of videos, stories, scenes, shots and frames as  $D_v$ ,  $D_s$ ,  $D_{sc}$ ,  $D_{sh}$ ,  $D_f$  respectively. The description of an entity is formed from several descriptors. For example, entity  $E_k$  can be described as follows:

$$D_{E_k} = \{d_{k1}, d_{k2}, d_{k3}, \dots\}$$

Where  $d_{k1}$ ,  $d_{k2}$ ,  $d_{k3}$ , ... are descriptors of the entity. Users can add descriptors to the entity.

**Definition 3. Utility.** Utility is the contribution of an entity. In other words, it explains how much work the entity does in representing the video content. We use descriptions of each entity to evaluate the utility. And by the utility function, we can get a series of utilities. Based on these utilities, we can finally generate video summarization.

### 2.2 The Formal Description of EDU Model

We describe EDU model as follows:

$$EDU = \{E, D, U, \varphi\} \quad (2)$$

where  $E$  is the entity set,  $D$  is the description set,  $U$  is the utility set, and  $\varphi$  is the relationship set of these sets.

Supposing  $E_v, E_{st}, E_{sc}, E_{sh}, E_f$  are the set of video entity, story entity, scene entity, shot entity, frame entity respectively, and has the following relationships:  $E_f \subseteq E_{sh} \subseteq E_{sc} \subseteq E_{st} \subseteq E_v \subseteq E$ , then EDU model can be described as follows:

$$U = \varphi(E_\alpha) = \varphi_3 \cdot \varphi_2 \cdot \varphi_1(E_\alpha) \quad (3)$$

where  $E_\alpha \subseteq E, \alpha \in \{f, sh, sc, st, v\}$ .  $\varphi_1, \varphi_2, \varphi_3$  means three types of operations, namely entity-to-entity, entity-to-description, description-to-utility. Then the above formulation can be described further in the following way:

$$E_\beta = \varphi_1(E_\alpha), \text{ where } E_\beta = \{e_1, e_2, \dots, e_n\}. \quad (4)$$

The equation reflects the process from entity to entity, we can think of  $\varphi_1$  as the operation of video segmentation or clustering. The equation shows that after operation  $\varphi_1$  on the entity  $E_\alpha$ , we get  $n$  entities, which can be expressed as  $E_\beta$ . For example, supposing  $E_\alpha$  is a video entity,  $E_\beta$  is a story entity, then the above equation means after story detection, we get  $n$  stories. Similarly, supposing  $E_\alpha$  is a shot entity, then after clustering operation  $\varphi_1$ , we can get the scene entity  $E_\beta$ .

Further, the process of entity to description can be described as:

$$D = \varphi_2(E_\beta), \text{ where } D = \{d_{ij} \mid 1 \leq i \leq n, 1 \leq j \leq m\}. \quad (5)$$

After the operation from entity to description, the description set  $D$  can be obtained. As each entity can be described with  $m$  descriptors, the descriptor set would have  $n \times m$  elements. Where  $d_{ij}$  means the  $j$ th description of the entity  $i$ .

Thirdly, generating utility from descriptions, which can be shown as follows:

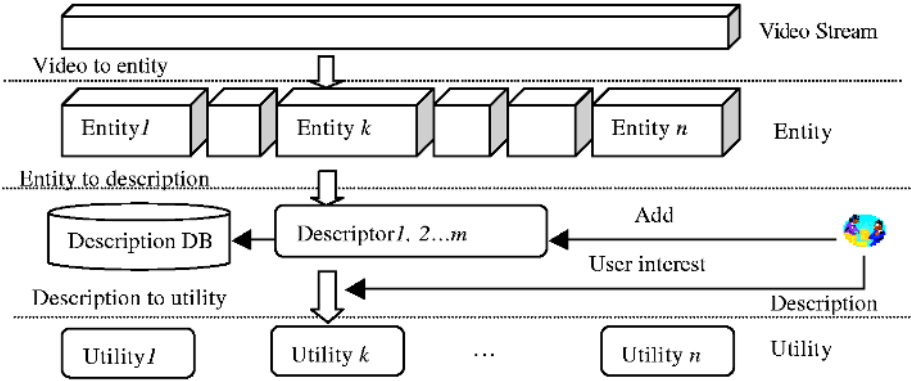
$$U = \varphi_3(D), \text{ where } U = (u_1, u_2, \dots, u_n)^T \quad (6)$$

where  $\varphi_3$  means the utility function. Supposing it is the simple weight sum function, then  $u_k = \sum_{j=1}^m w_j \cdot \overline{d_{ij}}$ , where  $\sum_{j=1}^m w_j = 1$ , and  $\overline{d_{ij}}$  is the normalization utility of the  $j$ th description of entity  $i$ .

### 2.3 The Structure of EDU Model

EDU model reflects the idea of generating summarization by the method of getting descriptions from entities and thus get utilities. (Ref. fig.1)





**Fig. 1.** Structure of EDU model

First, the original video streams should be segmented to get different levels of entities. According to different applications, different entities will be chosen. For example, to get summarizations of news stories, it is proper to choose shot as the basic entity; otherwise, to get summarization of a news topic, it's proper to choose news story as the basic entity.

Second, entities of different levels should be described automatically or half-automatically. For example, a shot entity can have many descriptors. If a face is detected in a shot, then add face descriptor to the shot entity, and save the information of occurrence time and position of the face. Other descriptors can also be added to a shot entity.

Finally, each entity's utility can be got by the utility function based on the descriptors. These series of utilities would be the basic measurement of the video summarization.

There are at least two advantages in the EDU model. First, it has a sharable description system, which is used to save preprocessed information. And the second, it has a utility function based on the descriptions, which is the measurement of video summarization. The first advantage shows its ease of expansibility, while the second one reflects users' interests in the summarization.

### 3 News Video Entity Description Based on EDU Model

As mentioned above, videos include five levels of entities, namely video stream, story, scene, shot and frame. As to news videos, the scene entity is neglected here. Different entities would be assigned with different descriptions.

Video entity description ( $D_v$ ) is the highest-level description, which describes the theme and the classification of a video, and answers questions like "what type is this video?"

News story entity description ( $D_{st}$ ) describes the topic and the clou of a story, tells users “what has happened?”

Shot entity description ( $D_{sh}$ ) describes the content of a shot from the physical and perceptive aspect. The content of shot description can be low-level features, such as duration, histogram, or can be high-level perceptive features, such as characters, scene types.

Frame entity description ( $D_f$ ) is the lowest level description, which seldom includes semantics, but mainly includes physical features such as colors, textures, etc.

All these descriptions form the description system.

## 4 News Story Summarization Method Based on EDU Model

News story is a basic unit for people to understand the news. Summarizing a news story is to give a quick view of the news story and preserve the important content. Considering the characters of news videos, we use shot as the basic entity for news story summarization.

### 4.1 From Shot Description to Shot Utility

Shot utility relies on shot description. It includes shot type utility, face utility and caption utility, etc. In this section, we will discuss how to calculate utilities from descriptors, and get the utility of a shot by the utility function mentioned in section 2.

**Shot Type Utility.** Each news story is formed from several shots. The similar shots can appear in a news story several times. For example, some scenes of the crowds may be edited to appear in a leader’s announcement. Obviously, the similar shots are redundant. They should be identified, and only the representative ones should be picked out

This can be accomplished by clustering similar shots. We adopt the key frame’s histogram recorded in the key frame descriptors to be a feature of a shot, and use the  $k$ -means clustering method to cluster similar shots.

Supposing we have got  $N$  clusters of shots in a news story, we define the weight of the  $i$  th cluster to be  $W_i$ [7]:

$$W_i = \frac{S_i}{\sum_{j=1}^N S_j} \quad (7)$$

Where  $S_i$  is the sum of duration of the  $i$ th cluster.

Generally, short and similar shots are not important, while those long and seldom appearing shots may be more important. This means with the rising of a cluster’s weights, the shot’s importance will decrease. So we define the importance of shot  $j$  of cluster  $k$  to be  $I_j$  (it is not a utility yet for it has not been normalized)[7]:

$$I_j = L_j \cdot \log \frac{1}{W_k} \quad (8)$$

Where  $L_j$  is the duration of shot  $j$ ,  $W_k$  is the weight of cluster  $k$ .

We conclude from the equation that with the rising of a cluster's weights, the importance of the shot is decreasing, while with the rising of duration of a shot, so does the importance. Then assign the utility of the shot with the highest importance in a news story as one, and the utility of the shot with the lowest importance to be zero. After normalization of the utilities, we can finally get the shot type utility  $\overline{d}_1$ .

**Face Utility.** The face descriptor of a shot includes the face picture itself, and the occurrence time of the face, the position of the face, etc. We can define their utility by the physical features.

Generally, users will pay more attention to a face in the central screen. We define the importance of a face as follows [6]:

$$I_{face} = \sum_{k=1}^N \frac{A_k}{A_{frame}} \times \frac{w_{pos}^j}{8} \quad (9)$$

Where  $I_{face}$  is the importance of the face in a shot,  $N$  is the total number of faces in a frame,  $A_k$  is the face's area in the frame  $k$ .  $A_{frame}$  is the area of the frame, and  $\frac{w_{pos}^j}{8}$  is the position weight.

We can see from the above equation that  $I_{face} \in [0,1)$ , and will be far less than 1 because the area of a face is only a small part of a frame. Similar to the shot type utility, we assign the face utility with the highest importance to be one, and the face utility with the lowest importance to be zero. After normalization of the utilities, we can finally get the face utility  $\overline{d}_2$ .

**Caption Utility.** Captions are frequent in news videos. They are titles of news stories or names of leaders. Captions are added for better understanding of news topics; only a little caption can be used as the description of the current shot such as names of the leaders.

As for the title captions, we think of them as the source of story description, and set the shot caption utility to be zero. For the annotation captions, we simply set their utility as 1. And for the shots with no caption, caption utility is obviously zero. And we mark the caption utility as  $\overline{d}_3$ .

**Other Utilities.** There are many other descriptors of shot entity besides type, face and caption. Users can add their needed descriptors such as motion. It should be mentioned here, each added descriptor should be assigned with its utility.

## 4.2 The Summarization Method Based on Changeable Threshold

We give equal weights to the above descriptors and can get each shot entity's utility by the utility function mentioned in section 2. Thus we get a series of utilities corresponding with each shot entity. We use the utility, which is a number between one and zero, to reflect the importance of entities in representing the whole news story (Ref. Fig. 2).

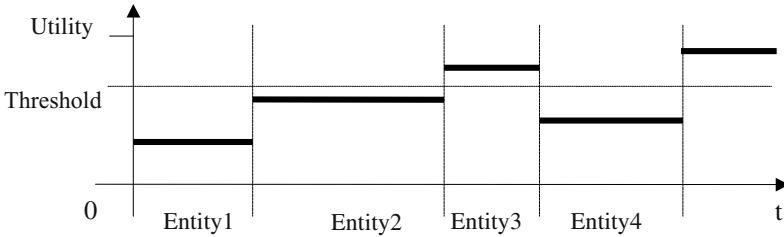


Fig. 2. Summarization method based on changeable threshold

As fig. 2 shows, the utility values are corresponding with the entities. The higher the utility is, the more important the entity is. The summarization is formed from those important entities. The threshold in fig. 2 is changeable with users' queries. For example, if we want to get 20% length of the original video, we can set the threshold from 1 to the point where all entities above the threshold can form 20% of the original video. Compared with the summarization method based on fixed threshold, this method is more representative and the length of summarization is easy to control.

## 5 Experiment Results

To evaluate the effectiveness of the EDU model, a group of usual descriptors were chosen for the experiment. The videos were from CCTV night news, Phoenix TV news and CCTV world report. The length of the original video was about 40 minutes. With the same condensation rate of 20%, we applied three different summarization methods to generate video skims, and invited fifteen students who hadn't seen the news before to evaluate the summarization results.

As fig. 3 shows, these three methods were: (1) Shot sampling and audio sampling; (2) Shot sampling and anchor voice choosing; (3) EDU model based summarization.

The first method was to choose 20% length of each shot from the start as the summarization. The second method was the same as the first; the difference was that it chose the voice of anchor shots. The third method was our proposed method based on EDU model; it also chose the voice of anchor shots.



generated summarization based on EDU model, and users could understand the news better.

## 6 Conclusions

In this paper, a unified video summarization model called EDU model was proposed. The model started from video entity, got utilities after descriptions, and finally generated video summarization by the utilities. As a general expandable model, EDU had a description scheme to save preprocessed information, and a utility function based on descriptions. Users could add their descriptors according to their needs. We could realize different levels of video summarizations. In the test, we chose summarizing news stories as an example, and proved the efficiency of the EDU model. In the future, more work will be done on the model, and we will try to apply this model to other summarization fields.

## References

1. Ying li, Tong Zhang, Daniel Tretter, An overview of video abstraction techniques, Image systems laboratory, HP Laboratory Palo Alto, HPL-2001-191, July 31st, 2001.
2. M. G. Christel, M. A. Smith, C. R. Taylor and D. B. Winkler, Evolving video skims into useful multimedia abstractions, Proc. of Conference on Human Factors in Computing Systems (CHI98), pp. 171-178, April 1998.
3. Hari Sundaram, Lexing Xie, Shih-Fu Chang, A utility framework for the automatic generation of audio-visual skims, ACM Multimedia'02, Dec.1-6, 2002, Juan-les-pins, France.
4. R. Lienhart, Dynamic video summarization of home video, Proc. of IS&T/SPIE, vol.3972, pp. 378-389, Jan. 2000.
5. R. Lienhart, S. Pfeiffer and W. Effelsberg, Video abstracting, Communications of the ACM, pp. 55-62, Dec. 1997.
6. Yu-Fei Ma, Lie Lu, Hong-jiang Zhang, Mingjing Li, A User Attention Model for Video Summarization, In Proc. of ACM Multimedia'02 December, 2002, Juan-les-Pins, France.
7. S. Uchihashi, J. Foote, A. Girgensohn and J. Boreczky, Video manga: generating semantically meaningful video summaries, ACM Multimedia'99, 1999.

# A News Video Mining Method Based on Statistical Analysis and Visualization

Yu-Xiang Xie, Xi-Dao Luan, Song-Yang Lao, Ling-Da Wu, Peng Xiao,  
and Zhi-Guang Han

Centre for multimedia Technology, National University of Defense Technology,  
Changsha, 410073, China  
{xyx89, xidaoluan}@sina.com, laosongyang@vip.sina.com

**Abstract.** In this paper, we propose a novel news video mining method based on statistical analysis and visualization. We divide the process of news video mining into three steps: preprocess, news video data mining, and pattern visualization. In the first step, we concentrate on content-based segmentation, clustering and events detection to acquire the metadata. In the second step, we perform news video data mining by some statistical methods. Considering news videos' features, in the analysis process we mainly concentrate on two factors: time and space. And in the third step, we try to visualize the mined patterns. We design two visualization methods: time-tendency graph and time-space distribution graph. Time-tendency graph is to reflect the tendencies of events, while time-space distribution graph is to reflect the relationships of time and space among various events. In this paper, we integrate news video analysis techniques with data mining techniques of statistical analysis and visualization to discover some implicit important information from large amount of news videos. Our experiments prove that this method is helpful for decision-making to some extent.

## 1 Introduction

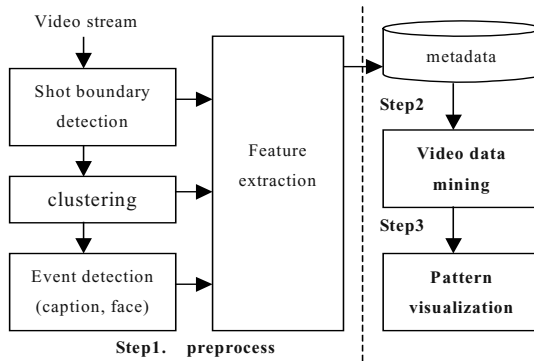
We come into contact with abundant news videos everyday, and can get a lot of information from them. Most news videos are actual and in time, so they are always helpful for people to make important decisions. For example, almost all investors have learned to find promising areas and industries by watching television to decide their investments. Some important news events would affect people's daily lives seriously, such as the epidemic SARS which happened not long ago did really have passive effect on various trades of the world, especially the tourism and people's living conceptions. When such events happen, we all wonder how and how long will it affect us. Will the situation become worse? All these suggest that news videos should not be one-off consumptions, but should be stored and analyzed seriously. By analyzing and mining large numbers of news video programs, we can find out valuable information to help people make their important decisions.

News video mining is a rising research field, which belongs to multimedia mining. Multimedia mining has become a hot research area in these years; it is an intercross

subject developed from various areas including multimedia database, data mining, information system, etc. and is defined as *the process of discovering the implicit and previously unknown knowledge or interesting patterns from a massive set of multimedia data*. Since the beginning of the first international conference of MDM/KDD in 2000, more and more scholars pay attention to multimedia mining. Many new conceptions, methods and framework theories of multimedia mining have been proposed [1], but most are confined to spatial data and image data mining. However, researches on video data mining are still in its infancy. Generally, there are three types of videos [1]: the produced, the raw, and the medical video. JungHwan [1] proposes a general framework and some methods of raw video mining. As to news video mining, much work is still to be done. Wijesekera [2] discusses the problem of applying traditional data mining methods to cinematic video mining; Kim [3] incorporates domain knowledge with audio-visual analysis in news video mining; Kulesh [4] proposes a personalized news video access method by the exposition of their PERSEUS project.

As mentioned above, there have been some efforts about video data mining in various fields. In this paper, we aim at discovering the implicit information in news videos. First, we extract and analyze statistically the news video content, and then propose two novel visualization methods: time-tendency graph and time-space distribution graph. Time-tendency graph is to reflect the tendencies of events, while time-space distribution graph is to reflect the spatial-temporal relationships among various events. These two visualization methods can be useful for decision-makers.

Figure 1 shows the flowchart of news video data mining. In this chart, we divide the process of news video mining into three steps: preprocess, video data mining (for example, statistical analysis), and pattern visualization.



**Fig. 1.** Flowchart of news video data mining

Corresponding to the above flowchart, we organize this paper as follows: Section 2 is the data preprocess of news video, including some preparation work such as shot boundary detection, clustering and event detection; section 3 is the statistical analysis of news stories; section 4 proposes two visualization methods, namely time-tendency graph and time-space distribution graph; Finally we give our concluding remarks in section 5.



## 2 Preprocess of News Video Data

As we know, video data is a kind of unstructured stream. In order to be recognized by computers, these data need to be preprocessed. In this stage, we will accomplish video syntactical segmentation and semantic annotation. In another word, we will segment news videos into meaningful units such as shots, scenes, stories, etc. We also try to get the semantic content of each unit in this stage.

Many content-based segmentation methods have been proposed. We adopt the method of comparing the color histograms of neighboring frames to decide the shot boundaries [6], and then use  $k$ -means method to cluster similar shots into scenes. By analyzing the time relationships between shots in a scene, we can get a series of news stories. These segmented stories will be the metadata for later video mining. So we will pay more attention to this basic unit.

Moreover, some semantic events in news videos may be interesting patterns to most users. Here, we mainly discuss two kinds of semantic events: caption event and face event. Some semantic events are periodic and repeated, for example, the caption events will happen periodically in the whole video stream and sometimes appear repeatedly in a news story. Some have not such obvious features of periodicity and repetition, such as face events in video streams.

We adopt the method proposed by Tang [9] to detect caption events in news videos. As to face detection events, we adopt the object detection method proposed in [5]. Then we set some basic rules to exclude small face events and reserve those feature face events.

Feature extraction is in parallel with the processes of shot boundary detection, clustering and event detection (Ref. Fig.1). In this process, we extract features of frames, shots, scenes, stories, faces, captions and some other description information, such as the occurrence time and space of news, which is achieved from speech recognition. All these features will be stored in the database.

## 3 Statistical Analysis of News Stories

After the preprocessing of news videos, we can get a series of news stories called metadata. But we still couldn't find out interesting patterns by arranging them in linear order only. To most decision-makers, they pay more attention to important news stories, which could give them profound impression and could effect greatly on their final decisions.

### 3.1 News Stories' Importance Model

Generally, important news would imply more information, and would always be the focus of the public. Therefore, it is necessary to model the news importance. Our news importance model would be considered from the following aspects:

**Sources.** It's understandable that news from authoritative TV stations would be more important and authoritative than those from local ones. Based on this, we divide news sources into five levels, namely world, country, province, city and county level. Then assign different importance values to different levels. The importance of the highest level is 1, and the others will decrease in size by 0.2 for each level. For example, news reported by the world level TV stations such as Reuters in British, CNN in America would be assigned with the highest importance value, while those reported by county TV stations would be set with the lowest one.

**Playtime.** News played in golden time would be more important than those not. In the same way, we assign different importance values to 24 periods of time in a day and assign higher importance values to golden time, such as A.M. 7, midday and 7 to 8 o'clock in the evening, etc. while news reported in midnights is always replay or unimportant one, and should be assigned with a lower importance value.

**Reported Frequency.** If the news is reported by several TV stations, or reported and tracked by a TV station several times, then we can believe that the news is important. Here we introduce the concept of duplicate in news. So-called duplicate means *the same news reported by different TV stations from different points of view*. Sometimes they are not visually similar with each other and can hardly be recognized by traditional algorithms. This forms the redundancy of video data and has bad influence on the search result. To solve this problem, Jaims [7] proposes a framework to detect duplicates. Based on this framework, we have designed a news duplicate detection method that integrates the algorithms of image and audio similarity matching, face matching, and voice scripts matching, and can detect news duplicates.

**Play Order.** In the same news program, news reported in the front would be more important. So the play order of news should be one of the important factors. This feature is more like the makeup of newspapers that the most important news is always arranged in the front page. We set the importance of the start time in one news program to be 1, and the importance of the end time in this news program to be 0. In this way, we build an inverse coordinate axis (Ref. Fig.2). Each news story's importance in the program can be computed by their starting time and is defined as  $P_i$  ( $P_i \in (0,1]$ ). For example, the thick line in Fig. 2 means a news story in a news program, according to its start time, its relative position  $P_i=0.7$ , which is also the news story's play order importance.

**Duration.** Generally, the duration of important news stories is longer than those not. For example, commonly, the news about the Iraq War lasts five to ten minutes, while other news lasts only two or less minutes. To some extent, the duration of news can suggest the importance of news stories.

**Feature Face.** After having detected the feature faces in section 2, we use the detected results to analyze the importance of news stories. For example, news appearing leaders' feature faces should be more important than those appearing only civilians'.



**Fig. 2.** Sketch map of relative position of a news story

According to all the characters mentioned above, we extract and save the necessary information of news stories, including play time, duration, play order, etc. And then propose the following news importance model.

Supposing  $I_s, I_p, I_d, I_o, I_t, I_f$  are importance measurement units, which mean the importance of a news story's source, playtime, play times, play order, duration and feature face importance respectively. And accordingly,  $w_1$  to  $w_6$  are six corresponding weights assigned to them. Then a simple linear combination model can represent the importance model of news stories as follows:

$$I = w_1 I_s + w_2 I_t + w_3 I_d + w_4 I_o + w_5 I_t + w_6 I_f \quad (1)$$

Where  $\sum_{i=1}^6 w_i = 1, I, I_i \in [0,1], i \in [s, t, d, o, l, f]$

### 3.2 Statistical Analysis of Time and Space

By using the importance model mentioned above, we can get a series of important news stories called topics. Since single news can't reflect the tendency and development of a topic, or discover the relationships of time and space between different news, it is necessary to adopt the method of statistical analysis to find them out.

Considering the factors of time, place, person and event of news, we focus on the factors of time and place. By analyzing these two factors, we can understand the relationships between the news more accurately.

To be mentioned here, our statistical analysis is performed on the same topic, for example, the topic of SARS. First, we calculate the event occurrence frequency along the time axis. As we know, most topics last a period of time, it maybe a week, a month or even longer. At different time point, relevant news of the same topic may happen more or less, and they form the developing process of the topic. We hope to find the implicit tendency among news stories by the statistical analysis along the time axis.

Given a topic  $I$ , supposing there are  $C_t$  news stories relevant to the topic happened on the same day  $t$ , and the importance of the news story  $i$  is  $I_i$ , then the topic's total importance on day  $t$  would be  $S(I, t)$ :

$$S(I, t) = \sum_{i=1}^{C_t} I_i \quad (2)$$

Thus, the time importance function of the topic is founded. For example, there are four pieces of relevant news about SARS happened on April the 19<sup>th</sup>, by computing each news importance ( $I_1, I_2, I_3, I_4$ ), we can conclude that the importance of SARS on April the 19<sup>th</sup> will be  $S_{(SARS, 4/19)} = I_1 + I_2 + I_3 + I_4$ . This importance can reflect the graveness degree of SARS on April the 19<sup>th</sup>.

In the same way, we can calculate the locations mentioned in the news and try to find the region centrality. In our experiment, we build a place database, in which many countries, regions and cities appeared in news are included. We obtain the occurrence place of each news by speech recognition and extracting place names.

Similar to the statistic of time axis, given a topic  $I$ , we calculate the topic's importance happened on the place  $p$  according to this formula:

$$S(I, p) = \sum_{i=1}^{C_p} I_i \quad (3)$$

Where  $S(I, p)$  means the topic's importance of the place  $p$ ,  $C_p$  means the number of the news relevant to the topic on place  $p$ , and  $I_i$  means the importance of the  $i$ th news story. For example, from April the 12<sup>th</sup> to April the 24<sup>th</sup>, there are 14 pieces of news related with SARS happened in Beijing. We compute each news importance ( $I_1, I_2, I_3, \dots, I_{14}$ ), and get the importance of SARS happened in Beijing  $S_{(SARS, Beijing)} = I_1 + I_2 + I_3 + \dots + I_{14}$ .

## 4 Visualization of Mined Patterns

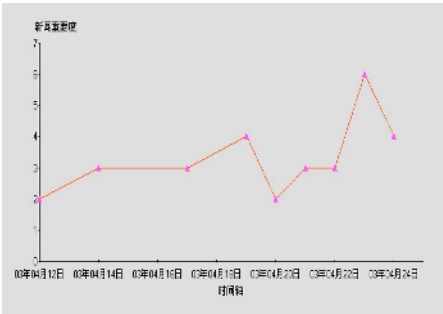
After the mining process of each stage, we have got some interesting patterns. Next, we will try to visualize these interesting patterns.

CMU university uses the method of timeline and map to visualize news videos [8], we are inspired to design two visualization methods: time-tendency graph and time-space distribution graph. Time-tendency graph is used to reflect the process of topics; time-space distribution graph reflects the relationships between time and space, and is helpful for decision-makers as a whole.

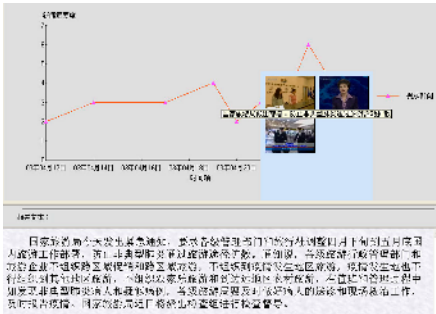
We collect the Night News of CCTV (an authoritative news program in China) from April the 12<sup>th</sup> to April the 24<sup>th</sup> in 2003 as experiment data set. Fig.3 is an example of time-tendency graph which indicates the time tendency of the topic SARS. The horizontal axis represents time, and the vertical axis represents the importance of the topic. We can see from Fig. 3 that from April the 12<sup>th</sup>, the news importance of SARS is arising, which indicates that SARS has become more and more serious in the world and media have paid more attention to it day by day.

In our system, if anyone is interested in the news stories on a certain day, he can browse the detailed news information by choosing the key frames. Fig. 4 is a user's browsing result of news stories relevant to SARS on April the 21<sup>st</sup>.

Fig. 5 and Fig. 6 are time-space distribution graph; they reflect a topic's current situations from the point of view of time and space. In another word, we add time axis to the map, thus we can get a topic's spatial distribution on a certain time point. The



**Fig. 3.** Time-tendency graph of the topic SARS



**Fig. 4.** Browsing the news stories relevant to SARS on April the 21<sup>st</sup>

red dots in the graph indicate the occurrence places of news stories; the sizes of them indicate the importance of the topic. By sliding the scroll bar below the map graph, we can choose to browse the time-space distribution of a day or a period of time. In Fig. 5, we choose April the 12<sup>th</sup>, and in Fig. 6, we choose April the 24<sup>th</sup>. Comparing these two graphs, the red dot located in Beijing in Fig.6 is bigger than in Fig.5, while the red dot located in GuangDong is quite the opposite. This indicates that from April the 12<sup>th</sup> to April the 24<sup>th</sup>, news reported about SARS in GuangDong has decreased, while is increased in Beijing. Then we can deduce that the situation of SARS in GuangDong is under control, while SARS in Beijing become worsening, which accords with the facts.



**Fig. 5.** Time-space distribution graph of SARS on April the 12<sup>th</sup>



**Fig. 6.** Time-space distribution graph of SARS on April the 24<sup>th</sup>

By adopting the methods we proposed above, we can finally draw the conclusions that in the period from April the 12<sup>th</sup> to April the 24<sup>th</sup>, SARS distributes mainly in Beijing and GuangDong. Among them, SARS in Beijing is more serious, and SARS in GuangDong is under control. According to these conclusions, decision-makers can adjust their investment and policy correspondingly to avoid losing or gain more profit.

## 5 Conclusions

In this paper, a news video mining method based on statistical analysis and visualization is proposed. According to news video's features, the method analyzes the news topics from two factors: time and space, discovers interesting patterns in the news, and designs two visualization methods: time-tendency graph and time-space distribution graph. Our primary experiments prove that this method is helpful for decision-making. News video mining is a representative and a promising research area in the field of multimedia mining. The framework and methods of news video mining is still in its infancy. We believe that news video mining would have great influence on various fields, such as information analysis, strategic decision and enterprise programming, etc.

## References

- [1] JungHwan Oh, Babitha Bandi, Multimedia data mining framework for raw video sequences, *Proc. of the 3rd International Workshop on Multimedia Data Mining (MDM/KDD'2002)*, July 23rd 2002, Edmonton, Alberta, Canada. pp: 1-10.
- [2] Duminda Wijsekera, Daniel Babara, Mining cinematic knowledge: Work in progress-An extended abstract, *Proc. of the 1st International Workshop on Multimedia Data mining (MDM/KDD'2000)*, August 20,2000,Boston, MA, USA. pp: 98-103.
- [3] Kim Shearer, Chitra Dorai, Svetha Venkatas, Incorporating domain knowledge with video and voice data analysis in news broadcasts, *Proc. of the 1st International Workshop On Multimedia Data Mining (MDM/KDD'2000)*, August 20,2000,Boston, MA,USA. Pp: 46-53.
- [4] Victor Kulesh, Valery A. Petrushin, Ishwark K.Sethi, The PERSEUS Project: Creating personalized multimedia news portal, *Proc. of the Second International Workshop on Multimedia Data mining (MDM/KDD'2001)*, San Francisco, USA, Aug. 26,2001, pp.31-37.
- [5] Rainer Lienhart, Jochen Maydt, An Extended Set Of Harr-like Features For Rapid Object Detection, *International Conference on Image Processing (ICIP'02)*, Rochester, New York, September, 22-25, 2002.
- [6] Kien A. Hua, JungHwan Oh, Khanh Vu, Non-linear approach to shot boundary detection, *IS&T/SPIE Conference on multimedia computing and networking*, 2001, pp: 1-12, Jan.22-25,2001, San Jose, CA.
- [7] Alejandro Jaims, Shih-Fu Chang, Alexander C. Loui, Duplicate detection in consumer photograph and news video, *Proc. of ACM Multimedia'02*, Dec.1-6, 2002, Juan-les-Pins, France.
- [8] Michael G. Christel, Alexander G. Hauptmann, Howard D. Wactlar, Tobun D. Ng, Collages as dynamic summaries for news video, *Proc. of ACM Multimedia'02*, Juan-les-Pins, France, December, 2002.
- [9] X. Tang, X. Gao, J. Liu, and H.J. Zhang, A Spatial-Temporal Approach for Video Caption Detection and Recognition, *IEEE Trans. On Neural Networks*, Special issue on Intelligent Multimedia Processing, July 2002, 13(4), pp.961-971

# Topic Threading for Structuring a Large-Scale News Video Archive

Ichiro Ide<sup>1,2</sup>, Hiroshi Mo<sup>2</sup>, Norio Katayama<sup>2</sup>, and Shin'ichi Satoh<sup>2</sup>

<sup>1</sup> Graduate School of Information Science, Nagoya University  
1 Furo-cho, Chikusa-ku, Nagoya-shi, Aichi 464-8601, Japan  
`ide@is.nagoya-u.ac.jp`

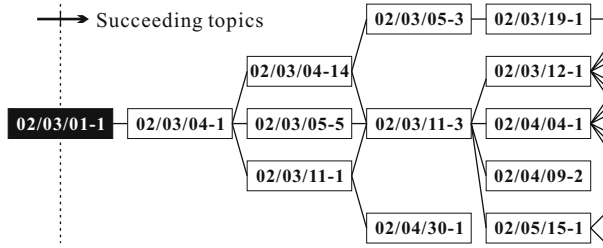
<sup>2</sup> National Institute of Informatics  
Research Organization of Information and Systems  
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan  
`{mo,katayama,satoh}@nii.ac.jp`

**Abstract.** We are building a broadcast news video archive where topics of interest can be retrieved and tracked easily. This paper introduces a structuring method applied to the accumulated news videos. First they are segmented into topic units and then *threaded* according to their mutual relations. A user interface for topic thread-based news video retrieval is also introduced. Since the topic thread structure is formed so that it has fewer number of emerging links from each topic than a simple link structure of related topics, it should lessen the tedious selection during a tracking process by a user. Although evaluation of the effect of threading and user study on the interface is yet to be done, we have found the interface informative to understand the details of a topic of interest.

## 1 Introduction

Broadcast video, especially news video contains a broad range of human activities which could be considered as a valuable cultural and social heritage. We are building a broadcast news video archive where topics of interest can be retrieved and tracked easily. The archive is supported by an automatic archiving system, a back-end contents analysis process, and a front-end user interface. In this paper, we will mainly focus on introducing the back-end contents analysis process, where the accumulated news videos are segmented into topic units and then *threaded* according to their mutual relations, and the front-end user interface.

The automatic archiving system captures and records broadcast news video streams including closed-caption texts (transcripts of audio), while the meta data are stored in a relational database. Currently, we have approximately 495 hours (312 GB of MPEG-1 and 1.89 TB of MPEG-2 format videos, and 23.0 MB of closed-caption texts) in the archive, obtained from a specific Japanese daily news program since March 16, 2001 (1,034 days in total). Each night, after the day's program is added to the database, the back-end contents analysis process will run. The process will be finished by the next morning so that a user can browse through the archive that reflects the topics added the previous night.



**Fig. 1.** Part of a topic thread structure extracted from the archive. Topics are labeled in the following format “Year/Month/Day-Topic#”.

**Topic thread structure in a news video archive.** A news video archive may seem merely an accumulation of video files recorded every day. Majority of previous works on news video structure analysis concentrated on segmenting a video stream into semantic units such as topics for retrieval (The most recent one: Yang *et al.* 2003). However, such retrieval is efficient only while the size of the archive remains relatively small. Once the archive grows larger, even the selection among the retrieved units becomes tedious for a user. Although several groups are dealing with news video archives of a comparable size with ours (Merlino *et al.* 1997; Christel *et al.* 1999), they do not look into the semantic relations between chains of topics. Works in Web structure mining is somewhat related to our work, but the existence of the chronological restriction makes our target substantially different.

We consider that linking semantically related topics in chronological order (*threading*) should be a solution to overcome this problem by providing a user with a list of topic threads instead of a whole list of individual topics. Once the topic thread structure of the entire archive has been revealed, it will no longer be a mere accumulation of video files, but a video archive where the threads complexly interweave related topics. Figure 1 shows an example of a topic thread structure starting from a topic of interest, which was actually obtained from the archive by the method proposed in this paper. As seen in the example, topic threads merge and diverge along time reflecting the transition of a story in the real world. Compare Fig. 1 with Fig. 2 which shows a simple link structure of related topics sorted chronologically. When providing a topic tracking interface by showing topics linked from a topic of interest, the fewer the number of links from each node exist, the less tedious the selection should be for a user.

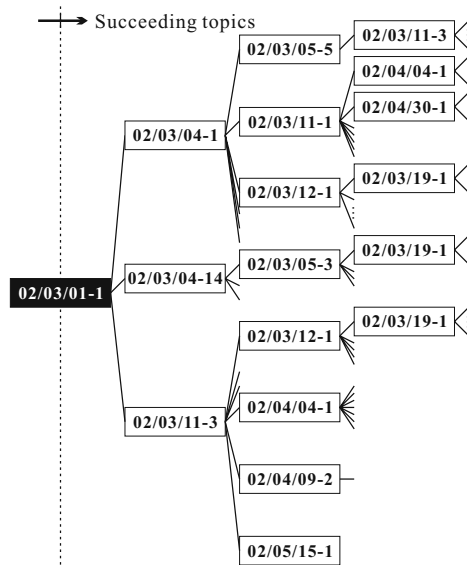
In the following Sections, the topic segmentation and threading methods are introduced, followed by the introduction of a thread-based browsing interface.

## 2 Topic Structuring

### 2.1 Topic Segmentation

A news topic is a semantic segment within a news video which contains a report on a specific incident. Compared with topic segmentation in general docu-





**Fig. 2.** Example of a simple topic link structure without threading.

ments, topic boundaries in a news video should be relatively clear, since a news video is naturally a combination of originally individual topics. Following this assumption, we will detect a topic boundary by finding a point between sentences where the keyword distributions within preceding and succeeding windows are distinctly different. The ideal window sizes are those when they are exactly the same with the actual topic lengths on both sides. Since the actual topic length is unknown beforehand, we will set elastic windows at each point between sentences to evaluate the discontinuity in various window sizes.

**Procedure.** The following steps were taken to detect a topic boundary from a closed-caption text broadcast simultaneously with the video. The closed-caption currently used is basically a transcript of the audio speech, though occasionally it is overridden by a source script or omitted when a superimposed caption is inserted in the video.

1. Apply morphological analysis to each sentence of a closed-caption text to extract compound nouns. A Japanese morphological analysis software, JUMAN (Kyoto Univ. 1999) was employed. Compound nouns were extracted since combination of adjacent nouns was considered as more distinctive to represent a topic, rather than a group of individual nouns.
2. Apply semantic analysis to the compound nouns to generate a keyword frequency vector for each semantic class (general, personal, locational / organizational, or temporal) per sentence ( $k_g, k_p, k_l, k_t$ ), which has frequencies as values. A suffix-based method (Ide *et al.* 2003) was employed for the analy-

sis, which classifies compound nouns both with and without proper nouns, according to suffix dictionaries for each semantic class.

3. At each boundary point between sentences  $i$  and  $i + 1$ , set a window size  $w$ , and measure the difference of keyword distributions between  $w$  preceding and succeeding sentences. The difference (or rather resemblance) is defined as follows, where  $i = w, w + 1, \dots, i_{max} - w$  when  $i_{max}$  is the number of sentences in a daily closed-caption text, and  $S = \{g, p, l, t\}$ .

$$R_{S,w}(i) = \frac{\sum_{m=i-w+1}^i \mathbf{k}_S(m) \cdot \sum_{n=i+1}^{i+w} \mathbf{k}_S(n)}{\left| \sum_{m=i-w+1}^i \mathbf{k}_S(m) \right| \left| \sum_{n=i+1}^{i+w} \mathbf{k}_S(n) \right|} \quad (1)$$

$$(2)$$

We set  $w = 1, 2, \dots, 10$  in the following experiment.

4. The maximum of  $R_{S,w}(i)$  among all  $w$  is chosen at each boundary as follows.

$$R_S(i) = \max_w R_{S,w}(i) \quad (3)$$

In preliminary observations, although most boundaries were correctly detected regardless of  $w$ , there was a large number of over-segmentation. We considered that taking the maximum should mutually compensate for over-segmentations at various window sizes, due to the following tendencies.

- Small  $w$ : Causes numerous over-segmentations, but has the advantage of showing significantly high resemblance within a short topic.
  - Large  $w$ : Does not always show high similarity within a short topic, but shows relatively high resemblance within a long one.
5. Resemblances evaluated in separate semantic attributes are combined as a weighted sum as follows.

$$R(i) = \sum_{S=\{g,p,l,t\}} a_S R_S(i) \quad (4)$$

Different weights are assigned to each semantic class under the assumption that certain attributes should be more important than others when considering topic segmentation especially in news texts.

Multiple linear regression analysis was applied to manually segmented training data (consists of 39 daily closed-caption texts, with 384 manually given topic boundaries) to determine the weights. The following weights were obtained as a result.

$$(a_g, a_p, a_l, a_t) = (0.23, 0.21, 0.48, 0.08) \quad (5)$$

The weights show that temporal nouns (*e.g.* today, last month) are not distinctive in the sense of representing a topic, where the other three, especially locational / organizational nouns act as distinctive keywords.

Finally, if  $R(i)$  does not exceed a certain threshold  $\theta_{seg}$ , the point is judged as a topic boundary.

6. To concatenate over-segmented topics, create a keyword vector  $\mathbf{K}_S$  for each topic, and re-evaluate the resemblances between adjoining stories  $i$  and  $j(=i+1)$  by the following function.

$$R(i, j) = \sum_{S=\{g,p,l,t\}} a_S \frac{\mathbf{K}_S(i) \cdot \mathbf{K}_S(j)}{|\mathbf{K}_S(i)| |\mathbf{K}_S(j)|} \quad (6)$$

As for  $a_S$ , the same weights as in Equation 5 were used.

If  $R(i, j)$  does not exceed a certain threshold  $\theta_{cat}$ , the topics are concatenated. This process continues until no more concatenation occurs.

**Experiment and evaluation.** The procedure was applied first to the training data used in Step 5. to define the thresholds ( $\theta_{seg} = 0.28, \theta_{cat} = 0.08$ ), and later to the entire archive ranging from March 16, 2001 to April 9, 2004 (1,034 days with 132,581 sentences in total). The whole process takes approximately 5 seconds per day on a Sun Blade-1000 workstation with dual UltraSPARC-III 750MHz CPUs and 2GB of main memory. As a result, 13,211 topics with more than two sentences were extracted. Topics with only one sentence (25,403 topics) were excluded since they tend to be noisy fragments resulting from over-segmentation.

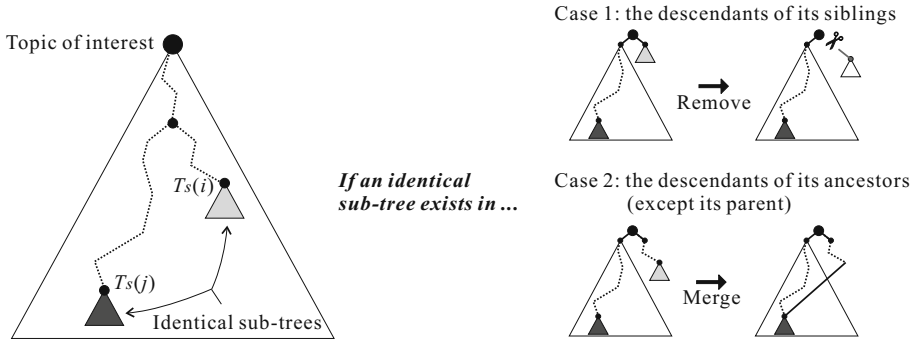
**Table 1.** Evaluation of topic extraction.

Condition	Both ends strict	One end strict / loose	Both ends loose
Recall	30.0%	34.6%	95.4%
Precision	28.5%	32.8%	90.5%

Evaluation was performed by applying the same procedure to manually segmented test data (consists of 14 daily closed-caption texts with 130 topics, set aside from the training data), which resulted as shown in Tab. 1. Boundaries were counted as correct if they matched exactly with the manually determined ones in ‘strict’ condition, and allowing  $\pm 1$  sentences in ‘loose’ condition. The ‘loose’ condition is acceptable for our goal since sentences at true boundaries tend to be short and relatively less informative regarding the main news content.

## 2.2 Topic Threading

A topic thread structure starting from a topic of interest is formed so that related topics are linked in chronological order. The difference between simply expanding related topics node by node as in Fig. 2 is that the proposed method replaces a link to a subordinate node if possible. The structure will therefore be rather *flat*; few branches at each node, and a long sequence of related topics instead. By this method, a topic that may eventually be reached in the tracking process will be pushed down so that a user needs not select among dozens of topics that he/she would never even need to see.



**Fig. 3.** Topic threading scheme.

**Procedure.** The thread structure is formed by the following algorithm.

1. Expand a topic link tree starting from the topic of interest so that it satisfies the following conditions.
  - a) Children are topics related to a parent, under the condition that their time stamps always succeed their parent's chronologically.
  - b) Siblings are sorted so that their time stamps always succeed their left-siblings' chronologically.

The resemblance between topics are evaluated by Equation 6. When  $R(i, j)$  exceeds a threshold  $\theta_{trk}$ , the topics are considered as related. This procedure forms a simple topic link tree such as the structure in Fig. 2.

Since evaluating numerous resemblances between various topics consumes too much time for real time processing in the user interface, resemblances between all possible topic pairs are evaluated beforehand. Currently, it takes roughly 1,400 seconds to add one new topic (comparing one topic against approximately 12,000 topics), which will keep on increasing as the archive grows larger.

2. For each sub-tree  $T_s(i)$ , if an identical sub-tree  $T_s(j)$  exists on the left-side, perform either of the following operations.
  - a) Remove  $T_s(i)$  if  $T_s(j)$  is a descendant of  $T_s(i)$ 's sibling.
  - b) Else, merge  $T_s(i)$  with  $T_s(j)$  if  $T_s(j)$  is a descendant of  $T_s(i)$ 's ancestor except its parent.

The sub-tree is removed in (a) instead of merging, to avoid creating a short-cut link without specific meaning. The removal and merger scheme is shown in Fig. 3. As a result of this operation, the thread structure will form a chronologically-ordered directed graph.

Note that this is in the case of forming a succeeding thread structure. A chronologically opposite algorithm is applied when forming a preceding thread structure.

To reduce computation time, the following conditions are applied in practice.

1. Pruning: Perform Step 2. whenever an identical story is found during the expansion of the tree in Step 1.

2. Approximation: Interrupt the expansion of the tree at a certain depth  $N_{trk}$ .

Although Condition 2. approximates the result, this will not affect much when referring to the direct children of the root (topic of interest) if  $N_{trk}$  is set to an appropriate value (We found  $N_{trk} = 3 \sim 5$  as sufficient in most cases).

### 3 Topic Thread-Based Video Retrieval

We built a topic retrieval interface, namely the “Topic Browser”, so that a user can browse through the entire news video archive by tracking up and down the topic threads. The interface consists of a “Topic Finder” and a “Topic Tracker”, which can be switched by tabs.

**The “Topic Finder”.** The “Topic Finder” is the portal to the tracking process; it retrieves a list of topics that contain the query term (Figure 4). A topic is represented by its meta data (date, time, topic ID), a thumbnail image (the first frame of the associated video), and an excerpt of the associated closed-caption text. The user can select the initial topic for tracking among the listed topics by actually viewing the video and associated close-caption text displayed on the right side of the interface.

**The “Topic Tracker”.** Once the user selects an initial topic, he/she will choose the “Topic Tracker” tab. To provide a list of topic threads starting from the initial topic, it performs the threading process as described in Sect. 2.2 on the fly (Figure 5). A topic thread is represented by the first topic in it, and key phrases that represent it so that the user can distinguish the difference with other threads. The first topics are selected to represent the threads since they were evaluated that they do not resemble each other, thus are considered as nodes where the topics diverge. The key phrases are noun sequences selected exclusively from the representative topic so that they do not overlap with those in other threads. The interface allows the user to set  $\theta_{trk}$ ,  $N_{trk}$  to adjust the number of threads to be displayed and the computation time.

The user will keep on selecting a topic over and over until he/she understands the details of the story during the tracking process, or finally finds a certain topic. The tracking direction is switchable so that it could go back and forth in time.

While “Topic Tracking” is in general a part of the “Topic Detection and Tracking (TDT) task” (Wayne 2000) in the TREC Conference, their definition of *tracking* and *detection* is somewhat static compared to what we are trying to realize in this interface. The point is that the proposed topic threading method extracts various paths that gradually track topics of interest that a user may follow requires our *tracking* to be more dynamic.

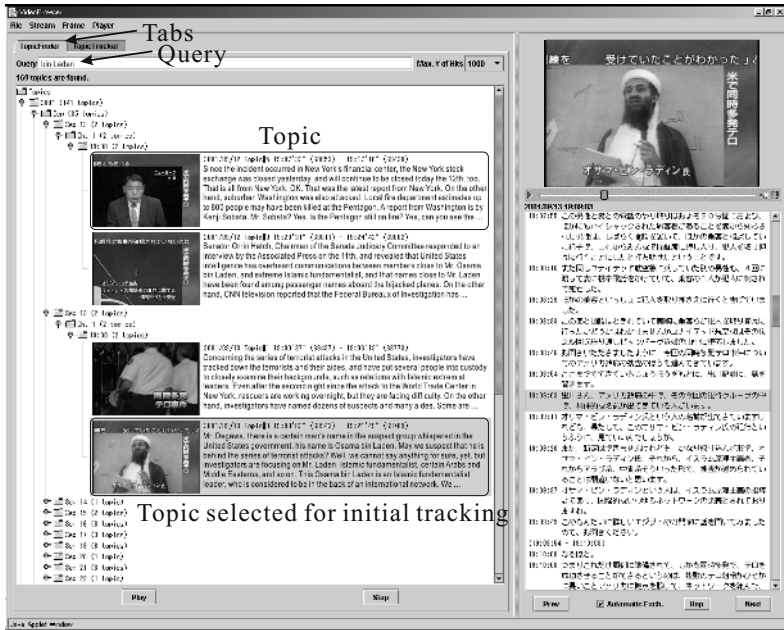


Fig. 4. The “Topic Finder” interface. Result of a query “Bin Laden”.

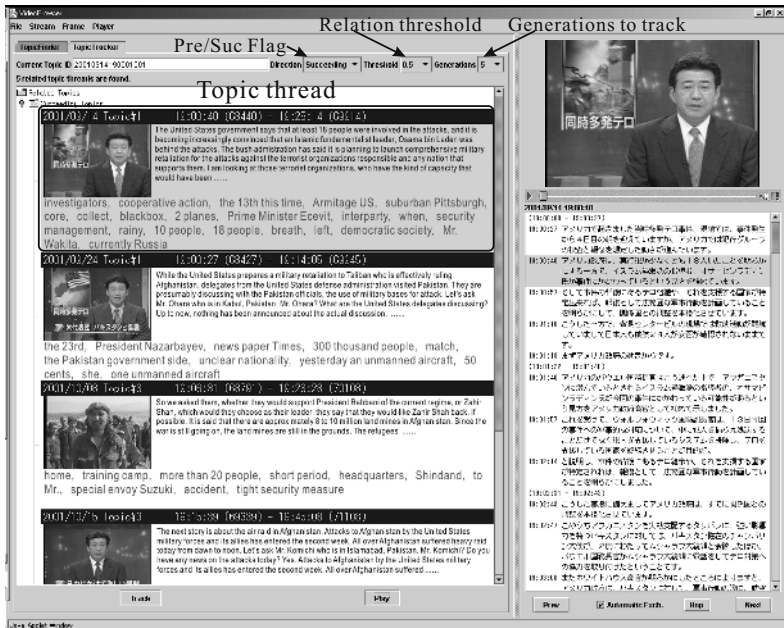


Fig. 5. The “Topic Tracker” interface.

## 4 Conclusions

We have proposed a method to reveal a topic-thread structure within a large-scale news video archive. The thread structure was applied to a video retrieval interface so that a user can track topics of interest along time. Since even the most relevant topic-based video retrieval method (Smeaton *et al.* 2003) considers the topic structure as simple links of related topics, the proposed approach is unique. Although precise evaluations and user studies are yet to be done, we have found the interface informative to understand the details of a topic of interest.

We will further aim at integrating image-based relations employing such methods as described in (Yamagishi *et al.* 2003) to link video segments that are related by semantics that could not be obtained from text. Precision of the tracking might be improved by refining the relation evaluation scheme by comparing a topic to a group of topics in a thread. A user study will also be performed to improve the retrieval interface after introducing relevance feedback in the tracking process, refining the keyword/thumbnailed selection scheme, and so on. Evaluation of the method to the TDT (Wayne 2000) corpus is an important issue, though the system will have to be adapted to non-Japanese transcripts.

**Acknowledgements.** Part of the work presented in this paper was supported by the Grants-in-Aid for Scientific Researches (15017285, 15700116) from the Japanese Society for the Promotion of Science, and the Japanese Ministry of Education, Culture, Sports, Science and Technology.

## References

- Christel, M. G., Hauptmann, A. G., Ng, T. D.: Collages as dynamic summaries for news video. Proc. 10th ACM Intl. Conf. on Multimedia (2002) 561–569
- Ide, I., Hamada, R., Sakai, S., Tanaka, H.: Compilation of dictionaries for semantic attribute analysis of television news captions. Systems and Computers in Japan **34**(12) (2003) 32–44
- Kyoto University: Japanese morphological analysis system JUMAN version 3.61.
- Merlino, A., Morey, D., Maybury, M.: Broadcast news navigation using story segmentation. Proc. 5th ACM Intl. Conf. on Multimedia (1997) 381–391
- Smeaton, A. F., Lee, H., O’Conner, N. E., Marlaw, S., Murphy, N.: TV news story segmentation, personalization and recommendation. AAAI Press Technical Reports, **SS-03-08** (2003) 60–65
- Wayne, C. L.: Multilingual topic detection and tracking: Successful research enabled by corpora and evaluation. Proc. 2nd Intl. Conf. on Language Resources and Evaluation **3** (2000) 1487–1493
- Yamagishi, F., Satoh, S., Hamada, T., Sakauchi, M.: Identical video segment detection for large-scale broadcast video archives. Proc. 3rd Intl. Workshop on Content-based Multimedia Indexing (2003) 135–142
- Yang, H., Chaisorn, L., Zhao, Y., Neo, S., Chua, T.: VideoQA: Question and answering on news video. Proc. 11th ACM Intl. Conf. on Multimedia (2003) 632–641

# What's News, What's Not?

## Associating News Videos with Words

Pinar Duygulu<sup>1</sup> and Alexander Hauptmann<sup>2</sup>

<sup>1</sup> Department of Computer Engineering, Bilkent University, Ankara, Turkey  
`pinar@cs.bilkent.edu.tr`

<sup>2</sup> Informedia Project, Carnegie Mellon University, Pittsburgh, PA, USA  
`alex@cs.cmu.edu`

**Abstract.** Text retrieval from broadcast news video is unsatisfactory, because a transcript word frequently does not directly ‘describe’ the shot when it was spoken. Extending the retrieved region to a window around the matching keyword provides better recall, but low precision. We improve on text retrieval using the following approach: First we segment the visual stream into coherent story-like units, using a set of visual news story delimiters. After filtering out clearly irrelevant classes of shots, we are still left with an ambiguity of how words in the transcript relate to the visual content in the remaining shots of the story. Using a limited set of visual features at different semantic levels ranging from color histograms, to faces, cars, and outdoors, an association matrix captures the correlation of these visual features to specific transcript words. This matrix is then refined using an EM approach. Preliminary results show that this approach has the potential to significantly improve retrieval performance from text queries.

## 1 Introduction and Overview

Searching video is very difficult, but people understand how to search text documents. However, a text-based search on the news videos is frequently errorful due to several reasons: If we only look at the shots where a keyword was spoken in a broadcast news transcript, we find that the anchor/reporter might be introducing a story, with the following shots being relevant, but not the current one. A speech recognition error may cause a query word to be mis-recognized while it was initially spoken during a relevant shot, but correctly recognized as the anchor wraps up the news story, leaving the relevant shot earlier in the sequence. Expanding a window of shots around the time of a relevant transcript word may boost recall, but is likely to also add many shots that are not relevant, thereby decreasing precision. Simple word ambiguities may also result in the retrieval of irrelevant video clips (*e.g.* is Powell, Colin Powell [secretary of state], Michael Powell [FCC chairman] or the lake?).

In this paper we lay out a strategy for improving retrieval of relevant video material when only text queries are available. Our first step segments the video into visually structured story units. We classify video shots as anchors [7], commercials [5], graphics or studio settings, or ‘other’ and use the broadcast video



editor's sequencing of these shot classes as delimiters to separate the stories. Separate classifiers were built to detect other studio settings and shots containing logos and graphics using color features. In the absence of labeled data, these latter two classifiers were built interactively. Using color features all shots were clustered and presented to the user in a layout based on a multi-dimensional scaling method. One representative is chosen from each cluster. Cluster variance is mapped into the image size to convey confidence in a particular cluster. Clusters with low variance are manually inspected and clusters corresponding to classes of interest are selected and labeled. Story boundaries are postulated between the classified delimiters of commercial/graphics/anchor/studio shots. Commercials and graphics shots are removed. Anchor and studio/reporter images are also deleted but the text corresponding to them is still used to find relevant stories.

The final step associates the text and visual features. On the visual side, we again create color clusters of the remaining (non-delimiter) shots. In addition we also take advantage of the results from existing outdoor, building, road and car classifiers. Finally, face detection results are also incorporated, grouping shots into ones with single faces, two faces, and three or more faces. On the text side, the words in the vocabulary are pruned to remove stop words and low frequency words. Then, co-occurrences are found by counting the associations of all words and all visual tokens inside the stories. The co-occurrences are weighted by the TF-IDF formula to normalize for rare or frequent words. Then a method based on Expectation Maximization is used to obtain the final association table.

We perform standard text retrieval on the query text, but instead of expanding a global window around the location of a relevant word, the story segmentation limits the temporal region in which shots relevant to the keyword may appear. All shots within a story containing a relevant query word are then re-ranked based on the visual features strongly correlated to this word based on the association table. This results in clear retrieval improvement over simplistic associations between a text word and the shot where it occurred. This approach also can help to find related words for a given story segment, or for suggesting words to be used with a classifier.

Our experiments were carried out on the CNN news data set of the 2003 TREC Video Track [11]. It contained 16650 shots as defined by a common shot segmentation, with one key-frame extracted for each shot.

## 2 Segmenting Broadcast News Stories Using Classes of Visual Delimiters

Our approach to segmentation relies on the recognition of visual delimiters inserted by the editors of the broadcasts. Specifically, we identify four types of delimiters: commercials, anchors, studio settings and graphics shots. While we had a large amount of training data for the first two delimiter types, we interactively built classifiers for the studio settings and graphics shots using a novel approach.

**Table 1. Top:** Number of shots detected and removed from each category. Remaining number of shots is 9497. **Bottom:** Number of correctly classified shots.

	anchors	commercials	graphics	in-studio
# elements	909	4347	1404	525
# correct	818 (90%)	4304 (99%)	1303 (93%)	456 (87%)

## 2.1 Commercials, Anchors, Studio, and Graphics Shots

In news video broadcasts, commercials are often inserted between news stories. For efficient retrieval and browsing of news, their removal is essential, as commercials don't contain any news material. To detect commercials, we use the approach in [5], which combines analysis of the repetitive use of commercials over time with their distinctive color and audio features.

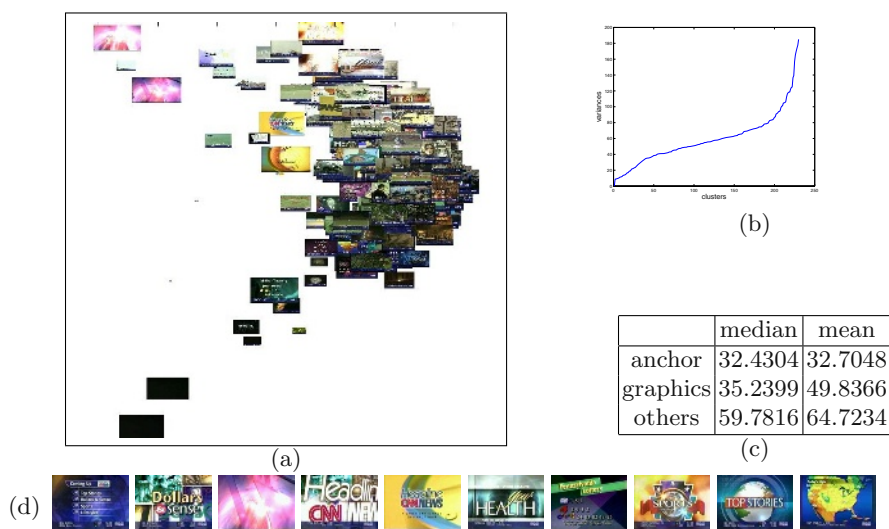
When searching for interesting broadcast news video material, detection and removal of anchor and reporter shots is also important. We use the method proposed in [7] to detect anchorpersons.

Even after the removal of commercial and anchor shots, the remaining video still contains more than the pure news story footage. Virtually all networks use a variety of easily identifiable graphics and logos to separate one story from another, or as a starting shot for a particular news category (for example characteristic logos appear before sports, weather, health or financial news) or as corporate self-identification such as the "CNN headline news" logo. While theoretically possible, it is quite tedious to manually label each of these graphics and build individual classifiers for them. However, these graphics appear frequently and usually have very distinctive color features and therefore can easily be distinguished.

In order to detect and remove these graphics, we developed a method which first clusters the non-anchor non-commercial shots using color features, and then provides an easy way to select clusters corresponding to these graphics. Similarly, shots that include studio settings with various anchors or reporters (apart from the straight anchorperson shots) can also be clustered, selected and removed to leave only shots of real news footage.

There are many ways to cluster feature sets, with differing quality and accuracy. Applying a K-means algorithm on feature vectors is a common method. However, the choice of K is by no means obvious. In this study, we use the idea of G-means [6] to determine the number of clusters adaptively. G-means clusters the data set starting from small number of clusters, C, and increases C iteratively if some of the current clusters fail the Gaussianity test (e.g., Kolmogorov-Smirnov test). In our study, 230 clusters were found using color based features. The specific color features were the mean and variance of each color channel in HSV color space in a 5\*5 image tessellation. Hue was quantized into 16 bins. Both saturation and value were quantized into 6 bins.

From each cluster a representative image is selected, which was simply the element closest to the mean, and these representatives are displayed using a Multi Dimensional Scaling (MDS) method. The layout is shown in Figure 1-a.



**Fig. 1.** (a) Representative images for the clusters. Size is inversely related to the variance of the cluster. (b) Distribution of variances for all clusters. (c) Mean and median variance values for selected graphics clusters, anchor clusters and others. (d) Example graphics clusters selected and later removed.

The size of the images is inversely related to the variance of the clusters. This presentation shows the confidence of the cluster. In-studio and graphics clusters tend to have less variance than other clusters, and therefore can be easily selected for further visual inspection. This process allows very quick review of the whole data set to label and remove the in-studio and graphics images. Table 1 shows the accuracy of our anchor, commercial, graphics and in-studio detection. Note that all detectors have an accuracy of 87% or higher, with commercials over 99%. We now remove all detected anchor, commercial, graphics and in-studio shots from the collection to leave only the shots which are related to news story footage. From the original 16650 shots, only 9497 shots were left as news story shots.

## 2.2 Segmenting with Visual Delimiters

To segment video news based on the visual editing structure, we devised a heuristic that uses the detected anchor, commercial, graphics and in-studio shots as delimiters. The algorithm is as follows:

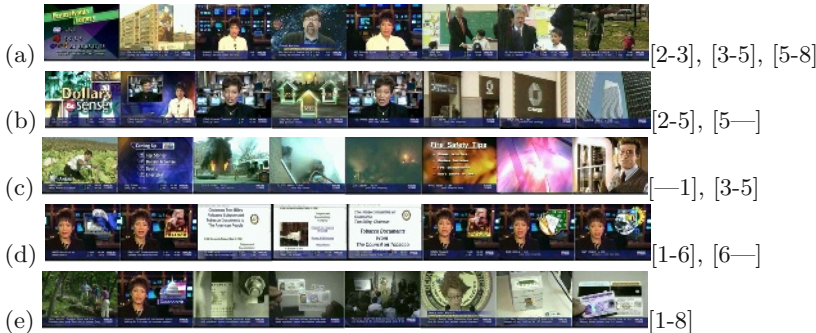
- Start a new story after a graphics or commercial shot.
- If there is a graphics or commercial in the next shot, then end the story.
- Start a new story with an anchor shot which follows a non-anchor shot.
- End a story with an anchor shot if the following is a non-anchor shot.

Figure 2 shows example segments obtained with the proposed algorithm. Most of the stories are correctly divided. Graphics create hard boundaries, while the anchor/reporter shots (and their associated text transcripts) are included

into both the preceding and following story segments. The reason is that an anchor usually finishes the previous story before starting another story. Without using textual information, the exact within-shot boundary cannot be accurately determined, nor can we tell if the anchor is only starting a story, but not finishing the previous one. Therefore, it is safer to add the anchor shots as part of both segments. This can be observed in Figure 2-d, where the iconic image in the right corner behind the anchor is same at the beginning and end of a story segment. However, in other cases, two news stories having different icons are merged into one. This problem could be solved with a more careful analysis of the icons behind the anchors or through text analysis. Other problems arise due to misclassification of the delimiter images. This may cause one story to be divided into two, or one story to begin late or end early, as in Figure 2-c and a delimiter may not be detected as in Figure 2-e. These problems again could be handled with a textual segmentation analysis.

To estimate the accuracy of our segmentation, 5 news broadcasts (1052 shots) were manually truthed for story segmentation. In the 5 broadcasts, 114 story segments were found and 69 segments were detected correctly. The errors are mostly due to incorrect splits or incorrect merges. In our case, the goal of story segmentation is to separate the news into parts for a better textual-visual association. Therefore, these incorrect segmentations actually are not very harmful or sometimes even helpful. For example, dividing a long story into two parts can be better since the further words are less related with the visual properties of the shots.

Other approaches that have been proposed for story segmentation are usually text-based. Integrated multimedia approaches have been shown to work well, however they incur great development and training costs, as a large variety of image, audio and text categories must be labeled and trained [2]. Informal analysis showed that our simple and cost-effective segmentation is sufficient for typical video retrieval queries [7].



**Fig. 2.** Example story segmentation results. (The segment boundaries are shown as [ $\langle$ starting shot $\rangle$  -  $\langle$ ending shot $\rangle$ ]. — is used to show that segment continues or starts outside of the selected shots.)

### 3 Associating Semantics with Visual Classifiers

The video is segmented into stories and the delimiters are removed from the video as described above. The data now consists of only the shots related to the news story and the transcript text related to that story. However, the specific image/text association is still unknown. Our goal is to associate the transcript words with the correct shots within a story segment for a better retrieval.

The problem of finding the associations can be considered as the translation of visual features to words, similar to the translation of text from one language to another. In that sense, there is an analogy between learning a lexicon for machine translation and learning a correspondence model for associating words with visual features.

In [3] association of image regions with keywords was proposed for region naming and auto-annotation using data sets consisting of annotated images. The images are segmented into regions and then quantized to get a discrete representation referred as 'blob tokens'. The problem is then transformed into translating blob tokens to word tokens. A probability table which links blob tokens with word tokens is constructed using an iterative algorithm based on Expectation Maximization [1] (For the details refer to [3]). A similar method is applied to link visual features with words in news videos [4] where the visual features (color, texture and edge features extracted from a grid) are associated with the neighbor words. However, this method have the problem of choosing the best window size for the neighborhood. In this study, we use the story segments as the basic units and associate the words and the visual features inside a story segment. Compared to [3], in our case, story segments take the place of images, and shots take the place of regions. In our study, also visual features are expanded with mid-level classifier outputs, which are called 'visual tokens'. The vocabulary is also processed to obtain 'word tokens'. The next section will give details how to obtain these tokens. Then, we describe a method to obtain the association probabilities and how they can be used for better retrieval.

#### 3.1 Extracting Tokens

We adapt some classifiers from Informedia's TREC-VID 2003 submission [7]. Outdoor, building, car and road classifiers are used in the experiments. Outdoor and road classifiers are based on the color features explained in the previous section and on texture and edge features. Oriented energy filters are used as texture features and a Canny edge detector is used to extract edges. The classifier is based on a support vector machine with the power=2 polynomial as the kernel function. Car detection was performed with a modified version of Schneiderman's algorithm [10]. It is trained on numerous examples of side views of cars. For buildings we built a classifier by adapting the man-made structure detection method of Kumar and Hebert[8] which produces binary detection outputs for each of 22x16 grids. We extracted 4 features from the binary detection outputs, including the area and the x and y coordinates of center of mass of the bounding box that includes all the positive grids, and the ratio of the number of positive

grids to the area of the bounding box. Examples, having larger values than the thresholds are taken as building images. For faces, Schneiderman’s face detector algorithm [10] is used to extract frontal faces. Here, shots are grouped into 4 different categories: no face (0), single face (1), two faces (2), three or more faces (3). Finally, color based clusters are also used after removing the clusters corresponding to in-studio and graphics shots. After removing 53 graphics and 17 in-studio clusters from 230 color clusters, 160 clusters are remained.

These classifiers are errorful. As shown in Table 2 removing the delimiters increases the accuracy of detections, but overall accuracy is very low. Our goal is to understand how visual information even if imperfect can improve retrieval results. As will be discussed later better visual information will provide better text/image association, therefore it is desirable to improve the accuracy of the classifiers, and also to create classifiers which are more specific and therefore more coherent.

On the text side, transcripts are aligned with shots by determining when each word was spoken. The vocabulary consists of only the nouns. Due to the errorful transcripts obtained from speech recognition, many incorrect words remain in the vocabulary. To remove stop words we only retained words occuring more than 10 times or less than 150 times, which cause the vocabulary to be pruned from originally 10201 words to 579 words.

**Table 2.** Classifier accuracies. **Before:** The original detection results on all the shots, **after:** after the removal of anchor, commercial and delimiter shots. Numbers show the number of shots detected correctly over all the detected shots. For outdoors due to the large number of images half of the data was truthed. Originally the number of detected outdoor shots was 5776 after removing anchors, delimiters and commercials.

classifier	outdoor	building	car	road
before	1419 / 4179 (34%)	126 / 924 (14%)	26 / 78 (33%)	71 / 745 (9%)
after	1000 / 2152 (46%)	101 / 456 (22%)	14 / 40 (35%)	40 / 421 (9%)

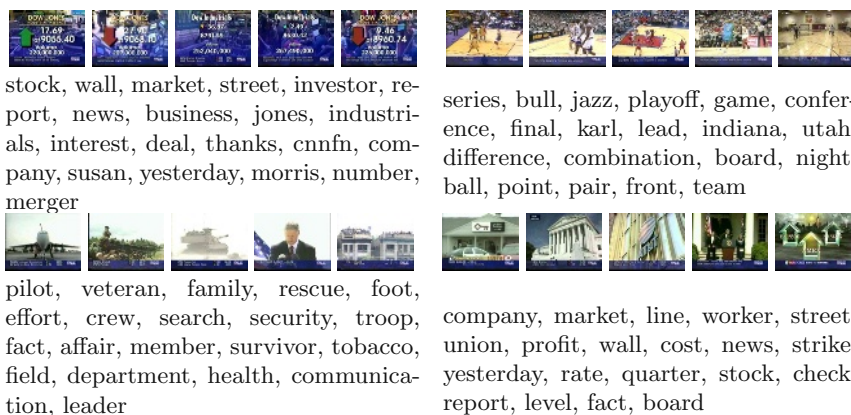
### 3.2 Obtaining Association Probabilities

Visual tokens are associated with word tokens using a “translation table”. The first step is finding the co-occurrences of visual tokens and words by counting the occurrence of words and visual tokens for each shot inside the story segments. The frequency of visual tokens are in a very large range which is also the case for words. In order to normalize the weights we apply tf-idf, which was used successfully in [9] for region-word association. After building the co-occurrence table, the final “translation table” is obtained using the Expectation-Maximization algorithm as proposed in [3]. The final translation table is a probability table which links each visual token with each word.

Figure 3 shows the top 20 words with the highest probability for some selected visual tokens. These tokens were chosen for their high word association probabilities. We observe that when a cluster is coherent the words associated

with it are closely related. Especially for sports and financial news this association is very clear. Building and road classifiers are relatively better than outdoor and car classifiers. The reason is that there are not many examples of cars, and the outdoor classifier is related to so many words due to number of outdoor shots.

The learned associations are helpful to do a better search. For a text based search, first the story segments which include the word are obtained. Then, instead of choosing the shot which is directly aligned with the query word, we choose the shot which has the highest probability of being associated with the query word. Figure 4 shows the search results for a standard time based alignment and after the words are associated with the shots using the proposed approach. For this example we choose only one shot from each story segment. Using sample queries for 'clinton' and 'fire', we find that 27 of 133 shots include Clinton using the proposed method (20% accuracy) while only 20 of 130 shots include him when the shots aligned with the text in time are retrieved (15% accuracy). For the 'fire' query, the numbers are 15/38 (40%) for the proposed approach and 11/44 (25%) for the time based approach.



**Fig. 3.** For three color tokens and for the building token, some selected images and the first 20 words associated with the highest probability.



**Fig. 4.** Search results **left:** for 'clinton', **right** for 'fire'. **Top:** Using shot text, **bottom:** the proposed method. While, the time based alignment produces unrelated shots (e.g anchors for clinton), the proposed system associates the words with the correct shots.

## 4 Conclusion

Association of transcripts with visual features extracted from the shots are proposed for a better text based retrieval. Story segmentation based on delimiters, namely anchor/commercial/studio/graphics shots is presented for extracting the semantic boundaries to link the shots and text. It is shown that by removing the delimiters and finding the associations it is possible to find the shots which actually correspond to the words. This method can also be used to suggest words and to improve the accuracy of classifiers. As observed in preliminary experiments, better visual tokens result in better associations. Having more specific classifiers may provide more coherent tokens. In the future we are planning to extend this work to motion information which can be associated with verbs. In this study only the speech transcript extracted was used. Text overlays can also be used for association.

**Acknowledgements.** This work was supported by the Advanced Research and Development Activity (ARDA) under contract numbers MDA908-00-C-0037 and MDA904-02-C-0451, and by the National Science Foundation (NSF) under Cooperative Agreement No. IIS-0121641. Also, we would like to thank Jia-yu Pan, Robert Chen, Rong Yan, Henry Schneiderman and Sanjiv Kumar for letting us to use their codes for detection and clustering.

## References

1. P.F. Brown and S. A. Della Pietra and V. J. Della Pietra and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation", *Computational Linguistics*, 19:2, 1993.
2. T.-S. Chua, Y. Zhao, L. Chaisorn, C.-K. Koh, H. Yang, H. Xu, "TREC 2003 Video Retrieval and Story Segmentation task at NUS PRIS", *TREC (VIDEO) Conference*, 2003.
3. P. Duygulu, K. Barnard, N. de Freitas, D. Forsyth, "Object recognition as machine translation: learning a lexicon for a fixed image vocabulary", *ECCV 2002*.
4. P. Duygulu and H. Wactlar "Associating video frames with text" *Multimedia Information Retrieval Workshop*, in conjunction with *ACM-SIGIR*, 2003.
5. P. Duygulu, M.-Y. Chen, A. Hauptmann, "Comparison and Combination of Two Novel Commercial Detection Methods", *ICME 2004*.
6. G. Hamerly and C. Elkan, "Learning the k in k-means", *NIPS 2003*.
7. A. Hauptmann et.al., "Informedia at TRECVID 2003:Analyzing and Searching Broadcast News Video", *TREC (VIDEO) Conference*, 2003.
8. S. Kumar and M. Hebert, "Man-Made Structure Detection in Natural Images using a Causal Multiscale Random Field", *CVPR*, 2003.
9. J.-Y. Pan, H.-J. Yang, P. Duygulu, C. Faloutsos, "Automatic Image Captioning", *ICME 2004*.
10. H. Schneiderman and T. Kanade, "Object detection using the statistics of parts", *International Journal of Computer Vision*, 2002.
11. TRECVID 2003, <http://www-nlpir.nist.gov/projects/tv2003/tv2003.html>



# Visual Clustering of Trademarks Using a Component-Based Matching Framework

Mustaq Hussain and John P. Eakins

School of Informatics,  
University of Northumbria at Newcastle, NE1 8ST, United Kingdom  
{mustaq.hussain, john.eakins}@unn.ac.uk

**Abstract.** This paper describes the evaluation of a new component-based approach to querying and retrieving for visualization and clustering from a large collection of digitised trademark images using the self-organizing map (SOM) neural network. The effectiveness of the growing hierarchical self-organizing map (GHSOM) has been compared with that of the conventional SOM, using a radial based precision-recall measure for different neighbourhood distances from the query. Good retrieval effectiveness was achieved when the GHSOM was allowed to grow multiple SOMs at different levels, with markedly reduced training times.

## 1 Introduction

The number and variety of image collections has risen rapidly over recent years which has led to increasing interest in automatic techniques for content based image retrieval (CBIR). Most CBIR techniques [1] compute similarity measures between stored and query images from automatically extracted features such as colour, texture or shape. Trademark image retrieval is currently an active research area [2], both because of their economic importance and because they provide a good test-bed for shape retrieval techniques. Two main approaches to trademark retrieval can be distinguished: the first based on comparison of features extracted from images as a whole (e.g. [3]), the second based on matching of image components (e.g. [4]). The latter approach appears to be more successful than the former [5].

Our previous work [6, 7] investigated the effectiveness of 2-D self-organizing maps (SOM) as a basis for trademark image retrieval and visualization. The aim of this study is to further investigate how the new component-based matching framework [7] scales up to larger image and query collections, and whether improvements in training times or retrieval effectiveness can be achieved using an adaptive SOM.

## 2 Self-Organizing Maps

The SOM [8] was selected as the basis for our approach to retrieval and visualization because it is an unsupervised topologically ordering neural network, a non-linear

projector, able to locally and globally order data and capable of accepting new data items without extensive re-training. It projects high dimensional data  $\mathbf{x}$  from  $n$ -D input space onto a low dimensional lattice (usually 2-D) of  $N$   $n$ -D neighbourhood connected (usually rectangular or hexagonal) nodes  $\{\mathbf{r}_i | i=1, \dots, N\}$  of weight  $\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{in}]$ .

A limitation of the SOM is that it has a static architecture, which has to be defined a-priori. To overcome this, several adaptive architectures have been proposed (e.g. [9], [10]). The Growing Hierarchical SOM (GHSOM) [11] is used in this study because of its regular shaped growing structure, adaptable growth to the data set, and rapid training times. It grows multiple layers where each layer is composed of independently growing SOMs. The training parameter  $\tau_1$  controls how the SOM grows horizontally (increasing in size) to allow it to represent the input space to a specific granularity, while parameter  $\tau_2$  controls the vertical expansion of a SOM node that represents too diverse input data to be represented by a new SOM at the next layer. This is done by training and growing each SOM independently, inserting rows and columns if node  $i$  with a mean quantization error ( $\text{mqe}_i$ )  $\frac{1}{|\mathbf{I}_k|} \sum_{\mathbf{x}_j \in \mathbf{I}_k} \|\mathbf{w}_i - \mathbf{x}_j\|$  has

$\text{mqe}_i > \tau_1 \text{mqe}_k$ , where the data subset  $\mathbf{I}_k$  is the input data set that best matches the parent node  $k$ , where  $\text{mqe}_k$  is from its parent node in the previous layer (for Layer 1 SOM its parent  $\text{mqe}_0$  is the mean error from the average input data). Once the SOM stops growing, nodes are examined for vertical insertion of a new SOM in the next layer if node  $i$  has  $\text{mqe}_i > \tau_2 \text{mqe}_0$ .

### 3 Component-Based Matching Framework

Images such as trademarks consist of a variable number of components. Hence the normal SOM training framework, based on a single fixed-length feature vector for each image, needs modification. Our component-based matching framework [7] treats each component as the basic unit for querying and retrieving, and uses these to train a topologically ordered component map. The map is subsequently queried to measure component similarities, which are combined to retrieve images, either as a 1-D ordered list or a 2-D cluster map. This two-stage process is defined below:

#### I. Database Construction:

Each test image  $\mathbf{T}^A$  and query image  $\mathbf{Q}$  is part of the image collection  $\mathbf{I}$ .

- (1) Each image  $\mathbf{T}^A$ , is segmented, into  $n$  components  $\mathbf{T}^A = \{\mathbf{t}^A | \mathbf{t}_1^A, \dots, \mathbf{t}_n^A\}$ , where  $n$  is variable and dependent on the image.
- (2) A suitable set of  $L$  image features measures  $f_{ij}^A$  is taken for each component  $\mathbf{t}_i^A$  creating a fixed-length component feature vector  $\mathbf{t}_i^A = \{f_{i1}^A, \dots, f_{iL}^A\}$ .
- (3) These fixed-length feature vectors  $\mathbf{t}^A$  are topologically ordered by training a SOM or GHSOM Component Map (CM) ( $\text{SOM}_{CM}$ ,  $\text{GHSOM}_{CM}$ ); similar components from different images should be close.

## II. Search and Display of Images:

- (4) The query image  $\mathbf{Q}$  is segmented and feature measures taken as in steps (1) and (2)  $\mathbf{Q} = \{\mathbf{q} \mid \mathbf{q}_1 = \mathbf{t}_1^Q, \dots, \mathbf{q}_m = \mathbf{t}_m^Q\}$ , with  $m$  fixed length components.
- (5) Create a 1-D ordered list of similar images (see [7] for full details).
- (6) Create a 2-D similarity cluster map:
  - (a) A Component Similarity Vector (CSV) is computed between each query component  $\mathbf{q}_i$  and test components  $\mathbf{t}^A$ , within neighbourhood radius  $r$  around each query component of the CM.
  - (b) Use these vectors to train a 2-D CSV SOM ( $\text{SOM}_{\text{CSV}}$ ) map.
  - (c) Display the topologically ordered  $\text{SOM}_{\text{CSV}}$  map.

The Component Similarity Vector (CSV)  $\mathbf{C}^A$ , of step (6), has  $m$  similarity measures, corresponding to the query's  $m$  components on the Component Map (CM) for the test image  $\mathbf{T}^A$  components:

$$\mathbf{C}^A = \begin{cases} c_i^A : \exp\left(-\frac{\|\mathbf{r}_i^Q - \mathbf{r}_c^A\|}{(2r^2)}\right) & \mathbf{t}_c^A \in \mathbf{N}_i \\ c_i^A : 0 & \mathbf{t}_c^A \notin \mathbf{N}_i \end{cases} \quad \forall i = 1 \dots m. \quad (1)$$

$\mathbf{C}^A = \{S(\mathbf{q}_1, \mathbf{t}^A), S(\mathbf{q}_2, \mathbf{t}^A), \dots, S(\mathbf{q}_m, \mathbf{t}^A)\}$  CSV with  $m$  components.

Each query component  $\mathbf{q}_i$  is searched a neighbourhood radius  $r$  around the CM for test image  $\mathbf{T}^A$  components. Each element  $c_i^A$  is a measure of how similar the test image  $\mathbf{T}^A$  is to the query component  $\mathbf{q}_i$ , therefore the whole vector  $\mathbf{C}^A$  is a measure of similarity to the whole query image.

This framework was tested successfully on a set of 4 test queries on a collection of 5268 trademark images [7], using fixed SOMs.

## 4 Present Experiments

Our previous work [7] established that good CSV SOM cluster maps can be generated from collections of up to 5000 images. Current experiments aim to answer questions about whether this framework will scale up to larger data sets and larger numbers of queries, and whether the use of GHSOMs has a beneficial effect on training times or retrieval effectiveness.

### 4.1 Image Collection and Feature Extraction

The images were all from a collection of 10745 abstract trademark images provided by the UK Trade Mark Registry – originally for the evaluation of the ARTISAN shape retrieval system [4]. Registry staff also assembled 24 query sets of similar

images, from this collection, which are used as independently established ground truth for retrieval.

All test images  $\mathbf{T}^A$  were segmented into boundary components  $\mathbf{t}^A$  by the method of [12]. For each component  $\mathbf{t}_i^A$ , 15 shape descriptors (which proved effective in comparative retrieval trials in that study) were extracted: (a) 4 "simple" shape features: relative area, aspect ratio, circularity and convexity; (b) 3 "natural" shape features proposed in [13]: triangularity, rectangularity and ellipticity; and (c) 8 normalized Fourier descriptors.

The result of segmenting the 10745 trademark images was a set of 85860 component feature vectors, each with 15 elements.

## 4.2 SOM and GHSOM Component Map Configuration

These 85860 component vectors  $\mathbf{t}_i^A$  were used to train a set of  $\text{SOM}_{CM}$  and  $\text{GHSOM}_{CM}$  maps (with different levels of horizontal and vertical expansion). Fixed sized  $\text{SOM}_{CM}$  maps reported here were of size  $150 \times 120$ ,  $175 \times 150$  and  $250 \times 200 = 50000$  nodes ( $\text{SOM}_{CM250 \times 200}$ ) to accommodate the 85860 components. The  $\text{SOM}_{CM150 \times 120}$  for example, during training had rough ordering parameters of  $\alpha_0=0.25$ ,  $N_0=40$  with 90,000 cycles and fine tuning parameters of  $\alpha_0=0.002$ ,  $N_0=6$  with 250,000 cycles and took 198.4 minutes which is summarized Table 1. All these hexagonal connected maps were all initialised in the same way. Larger maps were not used, as training times would have been excessive.

Table 2 shows the configuration of several  $\text{GHSOM}_{CM}$  maps grown by the selection of parameters  $\tau_i$  and  $\tau_j$ , that all grew from an initial  $2 \times 2$  Layer 1 SOM.

## 4.3 CSV SOM Training

With the trained  $\text{SOM}_{CM}$  &  $\text{GHSOM}_{CM}$ , an  $m$  element CSV  $\mathbf{C}^A(1)$  training vector was created for each test trademark image  $\mathbf{T}^A$ , for the query  $\mathbf{Q}$  with  $m$  components. These CSV vectors were next used to train a small  $12 \times 15 = 180$  node CSV SOM ( $\text{SOM}_{CSV12 \times 15}$ ) map. A small map was used because many CSV vectors were discarded, as many images had no components within the  $m$  query neighbourhoods. Finally, for visualisation, trademark images were positioned on their  $\text{SOM}_{CSV12 \times 15}$  best matching unit.

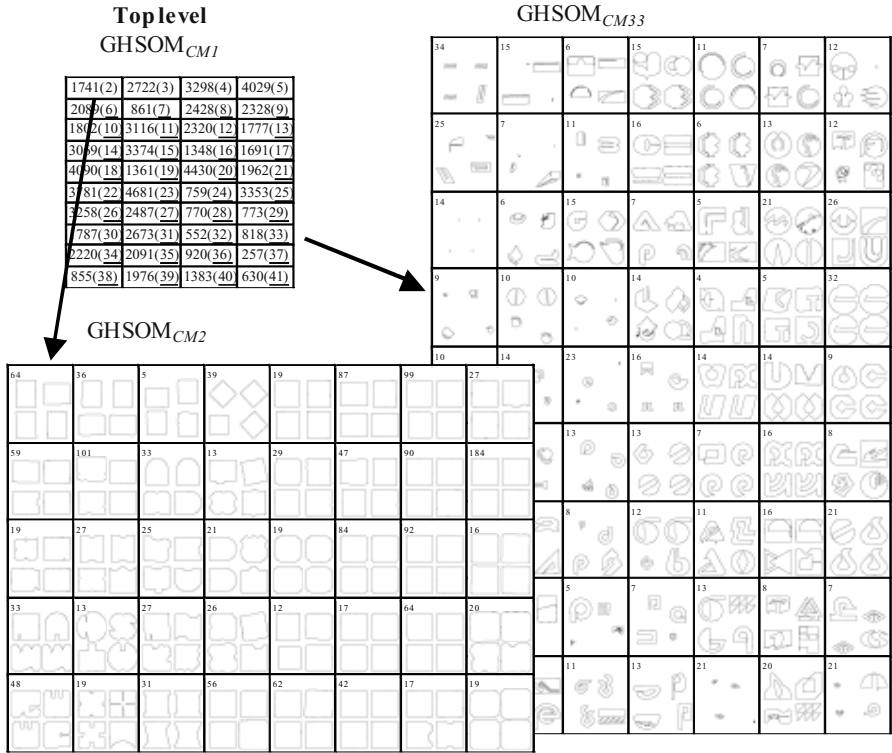
## 4.4 Retrieval Performance Evaluation

As with our previous work, retrieval effectiveness has been measured in terms of *precision*, the proportion of relevant items retrieved to the total retrieved, and *recall*, the proportion of relevant items retrieved. Like Koskela et al [14], we have modified this approach by computing radial precision  $P(r, \mathbf{Q})$  and recall  $R(r, \mathbf{Q})$  for query  $\mathbf{Q}$  for neighbourhood radius  $r$ , which operates as the maximum cut-off  $r=N_{co}$ . The average at radius  $r$ , over all  $N$  (in this study 24) query images is defined as  $P_{avg}(r) = \sum P(r, i)/N$  and  $R_{avg}(r) = \sum R(r, i)/N$  for  $\forall i = \{\mathbf{Q}_1, \dots, \mathbf{Q}_N\}$ , while the overall average for the whole

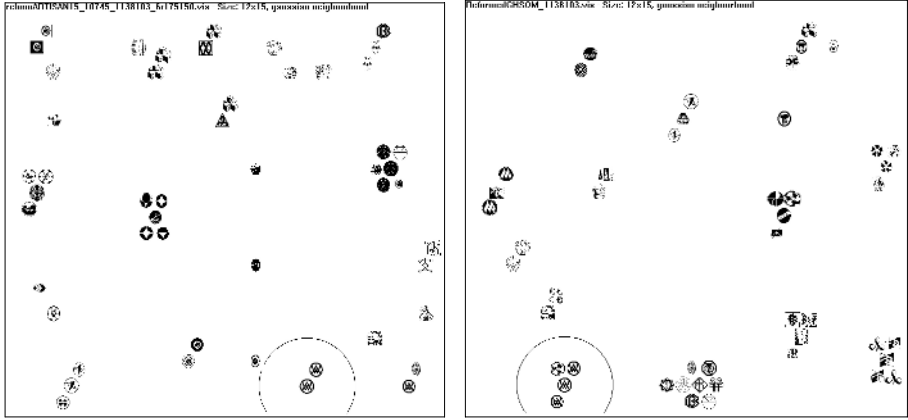
map is defined as  $P_{avg} = \sum P(r)/L$  and  $R_{avg} = \sum R(r)/L$  for  $\forall r=\{0,...,L-1\}$  out to radius  $L-1$ .

## 5 Visual Analysis of Results from GHSOM Component Map

Fig. 1 gives an example of a  $GHSOM_{CM}$  trained using the 85860 components. The top level (Level 1) map  $GHSOM_{CM1}$  has 40 nodes, all of which have been expanded into sub-maps. Each node shows the number of components that have been mapped onto that node; bracketed numbers give the sub-map number. The first sub-map  $GHSOM_{CM2}$ , at Level 2, is populated by square shaped components (four shown on each node for clarity). However,  $GHSOM_{CM33}$ , also on Level 2, is less uniform but still shows a large population of „C“ shaped components. This topological ordering is consistent with previous reports [5, 12] that the 15 shape measures used (see section 4.1) are good descriptors. Similar topological results were found for the large fixed size  $SOM_{CM}$  maps but because of the large number of nodes (e.g.  $175 \times 150 = 26250$  nodes) these maps would be too compressed to show clearly (see [7] for the smaller  $SOM_{CM60 \times 55}$ , from a 268 image set).



**Fig. 1.** Three  $GHSOM_{CM}$  maps are shown; with square shaped components on sub-map  $GHSOM_{CM2}$  and „C“ shaped components on sub-map  $GHSOM_{CM33}$ . Four components per node are shown on terminal nodes, and the number of components on each node



**Fig. 2.**  $SOM_{CSV12 \times 15}$  where CSV vectors were created from a (a)  $SOM_{CM175 \times 150}$  and (b)  $GHSOM_{CM}$  (query 1138103 has been circled)

## 6 Results and Analysis from Retrieval Experiments

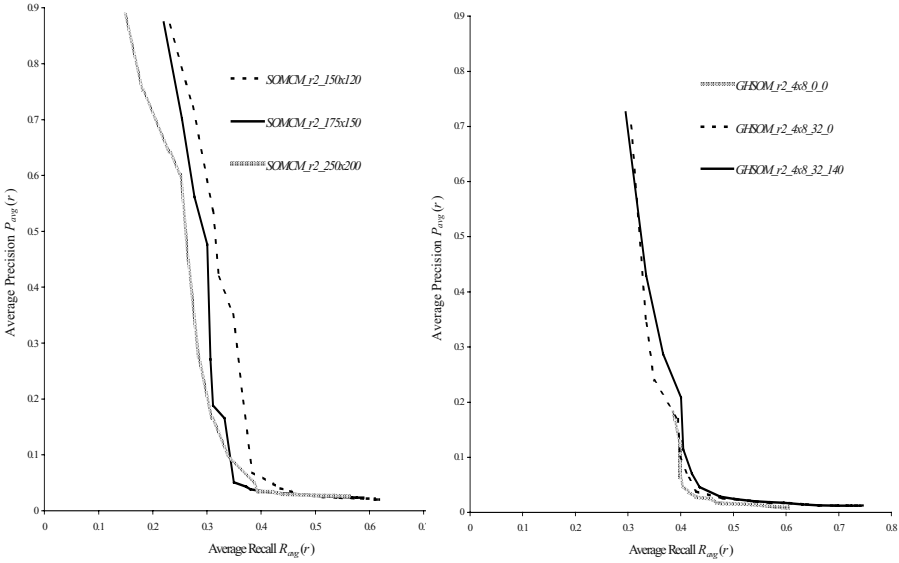
### 6.1 Qualitative Analysis of Visual Maps

The result of creating a set of CSV vectors and training two  $SOM_{CSV12 \times 15}$  maps for visualising retrieved trademarks is shown in Fig. 2, for the query 1138103 which has been circled and is shown surrounded by similar trademark images. Fig 2(a)'s CSV vectors were created by searching a fixed size  $SOM_{CM175 \times 150}$  with a neighbourhood radius of  $r=6$ . Fig. 2(b) was created by searching the trained  $GMSOM_{CM}$  of Fig. 1, again at radius  $r=6$ . These maps help to make clusters stand out, such as the query with expected result set members on Fig. 2(a) and the set of three in the lower left hand corner or the left-off-centre group. Qualitatively, clustering appears better using the  $SOM_{CM}$  of Fig. 2(a) than that using the  $GHSOM_{CM}$  of Fig. 2(b). Other query observations (from the set of 24) confirm that clusters are visually explicit on these 2-D  $SOM_{CSV}$  maps for this large collection of trademark images for both the large fixed  $SOM_{CM}$  and dynamic  $GHSOM_{CM}$ .

### 6.2 Quantitative Results and Analysis

The average precision-recall  $P_{avg}(r)-R_{avg}(r)$  graphs for all 24 query sets on the  $SOM_{CSV12 \times 15}$ , where the CSV vectors were created by searching various CM within a neighbourhood radius of  $r=2$  (from the query component) are shown in Fig. 3, from Table 1 & 2. Fig. 3(a) shows the fixed size  $SOM_{CM}$  ( $SOMCM\_r2\_<size>$ ), while Fig. 3(b) shows the  $GHSOM_{CM}$  with  $SOM_{CM}$  of size  $L1$  at Level 1, with  $L2/3$  number of  $SOM_{CM}$ s at Level 2/3 respectively ( $GHSOM\_r2\_<sizeL1>\_<L2>\_<L3>$ ).

Table 1 show six fixed size  $SOM_{CM}$  maps and their overall average precision-recall values on the  $SOM_{CSV12 \times 15}$  map using CSV vectors from the  $SOM_{CM}$  at radii 2, 4 and 6 neighbourhoods around the query components. Note that the training times involved were very long (1-3 hours).



**Fig. 3.** Average precision-recall graph of  $SOM_{CSV12x15}$  for (a) fixed size  $SOM_{CM}$  and (b)  $GHSOM_{CM}$  Component Maps within a search neighbourhood of 2, from Table 1 & 2

**Table 1.** CSV SOM Map’s overall average precision-recall  $P_{avg}-R_{avg}$ , at different radii  $r$  on  $SOM_{CM}$  Component Map and training times for all 24 queries

SOM <sub>CM</sub> size	Training Time (min)	SOM <sub>CSV12x15</sub> $P_{avg}-R_{avg}$ at different radii $r$ on SOM <sub>CM</sub>					
		$r=2$		$r=4$		$r=6$	
		$R_{avg}$	$P_{avg}$	$R_{avg}$	$P_{avg}$	$R_{avg}$	$P_{avg}$
150x120	60.3	0.459	0.184	0.536	0.115	0.572	0.090
175x150	83.6	0.418	0.190	0.496	0.121	0.541	0.087
250x200	198.4	0.392	0.221	0.459	0.159	0.507	0.116

Table 2 shows how by varying  $\tau_1$  and  $\tau_2$ , the  $GHSOM_{CM}$  can grow the  $SOM_{CM}$  on Layer 1 (always one SOM), and vary the number of  $SOM_{CM}$ s added in Layer 2 and 3. The overall average precision-recall of the  $SOM_{CSV12x15}$ , from the CSV vectors of the  $GHSOM_{CM}$  maps at radius 2 is given too. Note that the training times – often less than a minute – were much shorter than those for the single-layer fixed  $SOM_{CM}$  of Table 1. The first three  $GHSOM_{CM}$  maps with a large  $\tau_2$  only grew the Layer 1  $SOM_{CM}$ , and as  $\tau_1$  was increased its size got smaller and so did the training times. As  $\tau_2$  gets smaller (for fixed  $\tau_1$ )  $SOM_{CM}$  maps add SOM maps to the next level. For example, where  $\tau_1=0.05$  and  $\tau_2$  shrinks to 0.00004,  $SOM_{CM}$ s from Level 2 grew  $SOM_{CM}$ s at Level 3. SOMs at Level 2 were relatively quick to train as most were small and only a subset of the original training data set mapped onto their parent node, to train them. Similarly for Level 3 maps, which had smaller training sets.

**Table 2.** GHSOM Component Map structure for different parameter settings of  $\tau_1$  and  $\tau_2$  – size of Layer 1 map, number of maps in Layer 2 & 3, training time, and CSV SOM Maps’s overall average precision-recall, for radius  $r=2$  on GHSOM<sub>CM</sub>, for all 24 queries

	GHSOM Component Map parameters $\tau_1$ and $\tau_2$							
	$\tau_1$	0.0001	0.001	0.005	0.005	0.005	0.005	0.001
	$\tau_2$	0.04	0.04	0.04	0.004	0.0004	0.00004	0.0004
Layer 1 (always 1 SOM) size		22x26	8x12	4x8	4x8	4x8	4x8	8x12
Layer 2 no. maps		-	-	-	21	32	32	86
Layer 3 no. maps		-	-	-	-	-	140	-
Training Time		7m 2s	52s	21s	43s	48s	57s	2m 53s
SOM <sub>CSV</sub> $R_{avg}$		0.571	0.566	0.510	0.578	0.540	0.538	0.470
SOM <sub>CSV</sub> $P_{avg}$		0.059	0.038	0.034	0.076	0.097	0.110	0.216

Precision-recall profiles from Fig. 3 appear to agree with qualitative results from Fig. 2 that large fixed size SOM<sub>CM</sub> have better profiles than the small flat GHSOM<sub>CM</sub>, therefore are better at query retrieval and clustering. However, by expanding the poor profiled one layer GHSOM<sub>r2\_4x8\_0\_0</sub> (third GHSOM<sub>CM</sub> of Table 2) to a three layer GHSOM<sub>r2\_4x8\_32\_140</sub> (sixth GHSOM<sub>CM</sub> of Table 2), profiles are comparable to the fixed SOM<sub>CM</sub> and have been achieved in a shorter training period. One reason for this is that the GHSOM<sub>CM</sub> can describes components from crowded nodes onto specialised sub-maps as in Fig. 1, allowing precision to increase.

From Table 2, the third GHSOM<sub>CM</sub> with only one 4x8=32 node SOM<sub>CM</sub> at Level 1, expanded them all to Level 2 for the fifth GHSOM<sub>CM</sub>, therefore all image components were found at Level 2. However, the related fourth GHSOM<sub>CM</sub> only expanded 21 of its nodes, so this time component searching was conducted on two different levels. Results suggest that creating multiple maps improve precision, but the recall rates do not vary significantly.

## 7 Conclusions and Further Work

In this paper we investigated the ability of our component-based retrieval framework to handle a range of real queries with a collection of over 10000 trademark images. With large fixed SOM<sub>CM</sub>, retrieved SOM<sub>CSV</sub> trademark cluster maps were visually successful, but because of the larger number of images – and therefore components – the average precision-recall profiles were not been as good as in our previous study [7]. The multi-layered GHSOM<sub>CM</sub> also produced relatively good profiles in a shorter time (as can large single layered GHSOM<sub>CM</sub>, being similar to similar sized fixed SOM<sub>CM</sub>). However, they are not as good because relevant components were being distributed across multiple sub-maps, which can be missed when searching only one sub-map, as with Fig 1’s GHSOM<sub>CM2</sub> and GHSOM<sub>CM3</sub> with similar shaped square and circle components.

With the GHSOM<sub>CM</sub> it was found that nodes often had many components mapped onto them so the precision was low. This is being investigated by limiting the number of components that can share a node, therefore precision should go up. In cases where the neighbourhood radius extends beyond the edge of a sub-map, by extending the



search to cover adjacent sub-maps this too should increase recall, though at the expense of a possible loss of precision, as more image components are found.

We have also started to investigate whether this component-based framework can be applied to a different topological network – Generative Topological Mapping [15], a principled alternative to the SOM.

## References

1. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12) (2000), 1349-1380
2. Eakins, J.P.: Trademark image retrieval. In M. Lew (Ed.), *Principles of Visual Information Retrieval* (Ch 13). Springer-Verlag, Berlin (2001)
3. Kato, T.: Database architecture for content-based image retrieval. In: *Image Storage and Retrieval Systems*, Proc SPIE 2185 (1992), 112-123
4. Eakins, J.P., Boardman, J.M., Graham, M.E.: Similarity Retrieval of Trademark Images. *IEEE Multimedia*, 5(2) (1998), 53-63
5. Eakins, J.P., Riley, K.J., Edwards, J.D.: Shape feature matching for trademark image retrieval. Presented at The Challenge of Image and Video Retrieval (CIVR2003), *Lecture Notes in Computer Science*, 2728 (2003), 28-38
6. Hussain, M., Eakins, J.P., Sexton, G.: Visual Clustering of Trademarks Using the Self-Organizing Map. Presented at The Challenge of Image and Video Retrieval (CIVR 2002), *Lecture Notes in Computer Science*, 2383 (2002), 147-156
7. Hussain, M., Eakins, J.P.: Component Based Visual Clustering using the Self-Organizing Map. *Neural Networks*, submitted for publication
8. Kohonen, T.: *Self-Organizing Maps*, 3rd Ed. Springer Verlag, Berlin (2001)
9. Fritzke, B.: Growing cell structures - a self-organizing network for unsupervised and supervised learning, *Neural Networks*, Vol. 7, No. 9 (1994), page 1441-1460
10. Hodge, V.J., Austin, J.: Hierarchical Growing Cell Structures: TreeGCS. *IEEE Knowledge and Data Engineering*, 13(2) March/April (2001)
11. Dittenbach, M., Rauber, A., Merkl, D.: Uncovering hierarchical structure in data using the growing hierarchical self-organizing map. *Neurocomputing* 48. (2002) 199-216
12. Eakins, J.P., Edwards, J.D., Riley, J., Rosin, P.L.: A comparison of the effectiveness of alternative feature sets in shape retrieval of multi-component images. *Storage and Retrieval for Media Databases 2001*, Proc SPIE 4315 (2001), 196-207
13. Rosin, P.L.: Measuring shape: ellipticity, rectangularity and triangularity. *Proc of 15th International Conference on Pattern Recognition, Barcelona 1* (2000), 952-955
14. Koskela, M., Laaksonen, J., Laakso, S., Oja, E.: The PicSOM Retrieval System: Description and Evaluation. *Proceedings of The Challenge of Image Retrieval, Third UK Conference on Image Retrieval*, Brighton UK (2000)
15. Bishop, C.M., Svensén, M., Williams, C.K.I.: GTM: the generative topographic mapping. *Neural Computation*, 10(1) (1998), 215-234

# Assessing Scene Structuring in Consumer Videos

Daniel Gatica-Perez<sup>1</sup>, Napat Triroj<sup>2</sup>, Jean-Marc Odobez<sup>1</sup>,  
Alexander Loui<sup>3</sup>, and Ming-Ting Sun<sup>2</sup>

<sup>1</sup> IDIAP, Martigny, Switzerland

<sup>2</sup> University of Washington, Seattle WA, USA

<sup>3</sup> Eastman Kodak Company, Rochester NY, USA

**Abstract.** Scene structuring is a video analysis task for which no common evaluation procedures have been fully adopted. In this paper, we present a methodology to evaluate such task in home videos, which takes into account human judgement, and includes a representative corpus, a set of objective performance measures, and an evaluation protocol. The components of our approach are detailed as follows. First, we describe the generation of a set of home video scene structures produced by multiple people. Second, we define similarity measures that model variations with respect to two factors: human perceptual organization and level of structure granularity. Third, we describe a protocol for evaluation of automatic algorithms based on their comparison to human performance. We illustrate our methodology by assessing the performance of two recently proposed methods: probabilistic hierarchical clustering and spectral clustering.

## 1 Introduction

Many video browsing and retrieval systems make use of scene structuring, to provide non-linear access beyond the shot level, and to define boundaries for feature extraction for higher-level tasks. Scene structuring is a core function in video analysis, but the comparative performance of existing algorithms remains unknown, and common evaluation procedures have just begun to be adopted.

Scene structuring should be evaluated based on the nature of the content. (e.g. videos with “standard” scenes like news programs [3], or created with a storyline like movies [10]). In particular, home videos depict unrestricted content with no storyline, and contain temporally ordered scenes, each composed of a few related shots. Despite its non-professional style, home video scenes are the result of implicit rules of attention and recording [6,4]. Home filmmakers keep their interest on their subjects for a finite duration, influencing the time they spend recording individual shots, and the number of shots captured per scene. Recording also imposes temporal continuity: filming a trip with a non-linear temporal structure is rare [4]. Scene structuring can then be studied as a clustering problem, and is thus related to image clustering and segmentation [9,5].

The evaluation of a structuring algorithm assumes the existence of a ground-truth (GT) at the scene level. At least two options are conceivable. In the first-party approach, the GT is generated by the content creator, thus incorporating

specific context knowledge (e.g. place relationships) that cannot be automatically extracted by current means. In contrast, a third-party GT is defined by a subject not familiar with the content [4]. In this case, there still exists human context understanding, but limited to what is displayed. Multiple cues ranging from color coherence, scene composition, and temporal proximity, to high-level cues (recognition of objects/places) allow people to identify scenes in home video collections.

One criticism against third-party GTs is the claim that, as different people generate distinct GTs, no single judgement is reliable. A deeper question that emerges is that of consistency of human structuring of videos, which in turn refers to the general problem of perceptual organization of visual information<sup>1</sup>. One could expect that variations in human judgement arise both from distinct perceptions of a video scene structure, and from different levels of granularity in it [5]. Modeling these variations with an appropriate definition of agreement would be useful to compare human performance, and to define procedures to evaluate automatic algorithms. Similar goals have been pursued for image segmentation [5] and clustering [9], but to our knowledge work on videos has been limited.

We present a methodology to evaluate scene structuring algorithms in consumer videos. We first describe the creation of a corpus of 400 human-generated video scene structures extracted from a six-hour video database (Section 2). We then present a set of similarity measures that quantify variations in human perceptual organization and scene granularity (Section 3). The measures can be used to assess human performance on the task (Section 4), but they are also useful to evaluate automatic algorithms, for which we introduce an evaluation protocol (Section 5). Finally, the protocol is applied to compare the performance of two recent methods (Section 6). Section 7 provides some concluding remarks.

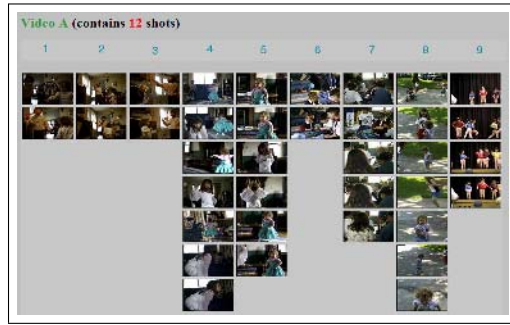
## 2 Video Scene Structure Corpus

### 2.1 Home Video Database

The data set includes 20 MPEG-1 videos, each with duration between 18-24 min. [4]. While relatively small (six hours), the set is representative of the genre, depicting both indoor (e.g. family gatherings and weddings), and outdoor (e.g. vacations) scenes. A manual GT at the shot level resulted in 430 shots. The number of shots per video substantially varies across the set (4-62 shots); see Fig. 2(a)).

---

<sup>1</sup> Perceptual organization is “a collective term for a diverse set of processes that contribute to the emergence of order in the visual input” [2], and “the ability to impose structural organization on sensory data, so as to group sensory primitives arising from a common underlying cause” [1]. In computer vision, perceptual organization research has addressed image segmentation, feature grouping, and spatio-temporal segmentation, among other problems, using theories from psychology (e.g. Gestalt).



**Fig. 1.** Scene structuring tool (detail). Each column of thumbnails represents a shot.

## 2.2 Tools for Scene Structuring

We define a video structure as composed of four levels (video clip, scene, shot, and subshot) [4]. Home video shots usually contain more than one appearance, due to hand-held camera motion, so subshots are defined to be intra-shot segments with approximately homogeneous appearance. A shot can then be represented by a set of key-frames (thumbnails) extracted from each of its subshots.

The amount of time required for human scene structuring is prohibitive when subjects deal with the raw videos. Providing a GUI with video playback and summarized information notably reduces the effort, but remains considerable for long videos due to video playing. In this view, we developed a GUI in which users were not displayed any raw videos, but only their summarized information (Fig. 1). Subshots and key-frames were automatically extracted by standard methods [4], and thumbnails were arranged on the screen in columns to represent shots. As pointed out in [8], images organized by visual similarity can facilitate location of images that satisfy basic requirements or solve simple tasks. In our case, the natural temporal ordering in video represents a strong cue for perceptual organization. In the GUI, a scene is represented by a list of shot numbers introduced by the user via the keyboard, so the scene structure is a partition of the set of all shots in a video, created by the user from scratch. Finding the scenes in a video depends of its number of shots, and it takes a couple of minutes in average.

## 2.3 The Task

A very general statement was purposely provided to the subjects at the beginning of the structuring process: “group neighboring shots together if you believe they belong to the same scene. Any scene structure containing between one and as many scenes as the number of shots is reasonable”. Users were free to define in their own terms both the concept of scene and the appropriate number of scenes in a video, as there was not a single correct answer. Following [5], such broad task was provided in order to force the participants to find “natural” video scenes.

## 2.4 Experimental Setup

**Participants.** A set of 20 university-level students participated in the experiments. Only two of the subjects had some knowledge in computer vision.

**Apparatus.** For each video in the database, we created the video structures as described in Section 2.2, using thumbnails of size  $88 \times 60$  pixels. We used PCs with standard monitors (17-inch,  $1024 \times 768$  resolution), running Windows NT4.

**Procedure.** All participants were informed about the purpose of the experiment and the GUI use, and were shown an example to practice. As mentioned earlier, no initial solution was proposed. Each person was asked to find the scenes in all 20 videos, and was asked to take a break if necessary. Additionally, in an attempt to refresh the subjects' attention on the task, the video set was arranged so that the levels of video complexity -as defined by the number of shots- was alternated. A total of 400 human-generated scene structures were produced in this way.

## 3 Measuring Agreement

If a unique, correct scene structure does not exist, how can we then assess agreement between people? Alternatives to measure agreement in image sets [9] and video scenes [4] have been proposed. By analogy with natural image segmentation [5], here we hypothesize that variations in human judgement of scene structuring can be thought of as arising from two factors: (i) distinct perceptual organization of a scene structure, where people perceive different scenes altogether, so shots are grouped in completely different ways, and (ii) distinct granularity in a scene structure, which generates structures whose scenes are simply refinements of each other. We discuss both criteria to assess consistency in the following subsections.

### 3.1 Variations in Perceptual Organization

Differences in perceptual organization of a scene structure, that is, cases in which people observe completely different scenes, are a clear source of inconsistency. A definition of agreement that does not penalize granularity differences was proposed in [5] for image segmentation, and can be directly applied to video partitions. Let  $S_i$  denote a scene structure of a video (i.e., a partition of the set of shots, each assigned to one scene). For two scene structures  $S_i, S_j$  of a  $K$ -shot video, the *local refinement error* ( $LRE$ ) for shot  $s_k$ , with range  $[0, 1]$ , is defined by

$$LRE(S_i, S_j, s_k) = ||R(S_i, s_k) \setminus R(S_j, s_k)|| / ||R(S_i, s_k)||, \quad (1)$$

where  $\setminus$  and  $||\cdot||$  denote set difference and cardinality, respectively, and  $R(S_i, s_k)$  is the scene in structure  $S_i$  that contains  $s_k$ . On one side, given shot  $s_k$ , if  $R(S_i, s_k)$  is a proper subset of  $R(S_j, s_k)$ ,  $LRE = 0$ , which indicates that the first scene is a refinement of the second one. On the other side, if there is no overlap between the two scenes other than  $s_k$ ,  $LRE = (||R(S_i, s_k)|| - 1) / ||R(S_i, s_k)||$ , indicating an inconsistency in the perception of scenes.

To obtain a global measure,  $LRE$  has to be made symmetric, as  $LRE(S_i, S_j, s_k)$  and  $LRE(S_j, S_i, s_k)$  are not equal in general, and computed over the entire video. Two overall measures proposed in [5] are the *global* and *local consistency errors*,

$$GCE(S_i, S_j) = \frac{1}{K} \min \left\{ \sum_k LRE(S_i, S_j, s_k), \sum_k LRE(S_j, S_i, s_k) \right\}, \quad (2)$$

$$LCE(S_i, S_j) = \frac{1}{K} \sum_k \min \{LRE(S_i, S_j, s_k), LRE(S_j, S_i, s_k)\}. \quad (3)$$

To compute  $GCE$ , the  $LRE$ s are accumulated for each direction (i.e. from  $S_i$  to  $S_j$  and vice versa), and then the minimum is taken. Each direction defines a criterion for which scene structure refinement is not penalized. On the other hand,  $LCE$  accumulates the minimum error in either direction, so structure refinement is tolerated in any direction for each shot. It is easy to see that  $GCE \geq LCE$ , so  $GCE$  constitutes a stricter performance measure than  $LCE$  [5].

### 3.2 Variations in Structure Granularity

The above measures do not account for any differences of granularity, and are reasonably good when the number of detected scenes in two video scene structures is similar. However, two different scene structures (e.g. one in which each shot is a scene, and one in which all shots belong to the same scene) produce a zero value for both  $GCE$  and  $LCE$  when compared to any arbitrary scene structure. In other words, the concept of “perfect agreement” as defined by these measures conveys no information about differences of judgment w.r.t. the number of scenes. In view of this limitation, we introduce a revised measure that takes into account variations on the number of detected scenes, by defining a weighted sum,

$$GCE'(S_i, S_j) = \alpha_1 GCE(S_i, S_j) + \alpha_2 C(S_i, S_j), \quad (4)$$

where  $\sum_i \alpha_i = 1$ , and the correction factor  $C(S_i, S_j) = \frac{|N(S_i) - N(S_j)|}{N_{max}}$ , where  $N(S_i)$  is the number of scenes detected in  $S_i$ , and  $N_{max}$  is the maximum number of scenes allowed in a video ( $K$ ). A similar expression can be derived for  $LCE'$ .

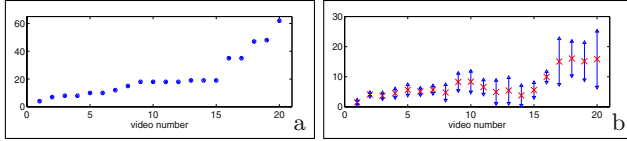
## 4 Human Scene Structuring

The discussed measures were computed for all pairs of human-generated scene structures for each video in the data set. Note that, as shots are the basic units, partitions corresponding to different videos are not directly comparable. Fig. 3(a) shows the distributions of  $GCE$  and  $LCE$  between pairs of human-generated scene structures of the same video. All distributions show a peak near zero, and the error remains low, with means shown in Table 1. It is also clear that  $GCE$  is a harder measure than  $LCE$ . Given the measures that only penalize differences

in perceptual organization, people produced consistent results on most videos on the task of partitioning them into an arbitrary number of scenes.

However, the variation in performance with respect to the number of detected scenes -not directly measured by  $GCE/LCE$ - is considerable. Fig. 2(b) displays the mean and standard deviation of the number of detected scenes for each video. The videos are displayed in increasing order, according to the number of shots they contain (Fig. 2(a)). As a general trend, videos with more shots produce larger variation in the number of detected scenes. Referring to Fig. 3(a), the strong peaks near zero are somehow misleading, as it is obvious that human subjects did not produce identical scene structures. The distribution of the new performance measures ( $GCE'$  and  $LCE'$ ) for weights  $\alpha_1 = 0.85, \alpha_2 = 0.15$  are shown in Fig. 3(b). The weights were chosen so that the weighted means of  $GCE$  and  $C$  approximately account for half of the mean of  $GCE'$ . For the new measures, the distributions no longer present peaks at zero. The errors are higher, as they explicitly penalize differences in judgement regarding number of scenes.

Overall, given the small dataset we used, the results seem to suggest that (i) there is human agreement in terms of perceptual organization of videos into scenes, (ii) people present a large variation in terms of scene granularity, and (iii) the degree of agreement in scene granularity depends on the video complexity.



**Fig. 2.** (a) Number of shots per video in the database (in increasing order); (b) mean and standard deviation of number of scenes detected by people for each video.

## 5 Evaluation Protocol

To evaluate an automatic method by comparing it to human performance, two issues have to be considered. First, the original measures ( $GCE/LCE$ ) are useful for comparison when the number of scenes in two scene structures is similar. This is convenient when the number of scenes is a free parameter that can be manually set, as advocated by [5]. However, such procedure would not measure the ability of the algorithm to perform model selection. For this case, we think that the proposed measures ( $GCE'/LCE'$ ) are more appropriate. Second, the performance of both people and automatic algorithms might depend on the individual video complexity.

In this view, we propose to evaluate performance by the following protocol [7]. For each video, let  $S_A$  denote the scene structure obtained by an automatic algorithm, and  $S_j$  the  $j$ -th human-generated scene structure. We can then compute  $GCE'(S_A, S_j)$  for all people, rank the results, and keep three measures: minimum, median, and maximum, denoted by  $GCE'_{min}(S_A, S_j)$ ,  $GCE'_{med}(S_A, S_j)$ ,

and  $GCE'_{max}(S_A, S_j)$ , respectively. The minimum provides an indication of how close an automatic result is to the nearest human result. The median is a fair performance measure, which considers all the human responses while not being affected by the largest errors. Such large errors are considered by the maximum. An overall measure is computed by averaging the  $GCE'$  measures over all the videos. To compute the same measures among people, for each video, the three measures are computed for each subject against all others, and these values are averaged over all subjects. The overall performance is computed by averaging over all videos. To visualize performance, it is useful to plot the distributions of  $GCE'$  and  $LCE'$ , obtained by comparing automatic and human-generated scene structures, as in Fig. 3. Finally, to compare two algorithms, the described protocol can be applied to each algorithm, followed by a test for statistical significance.

## 6 Assessing Automatic Algorithms

### 6.1 The Algorithms

We illustrate our methodology on two recently proposed algorithms based on pair-wise similarity. For space reasons, we briefly describe the algorithms here.

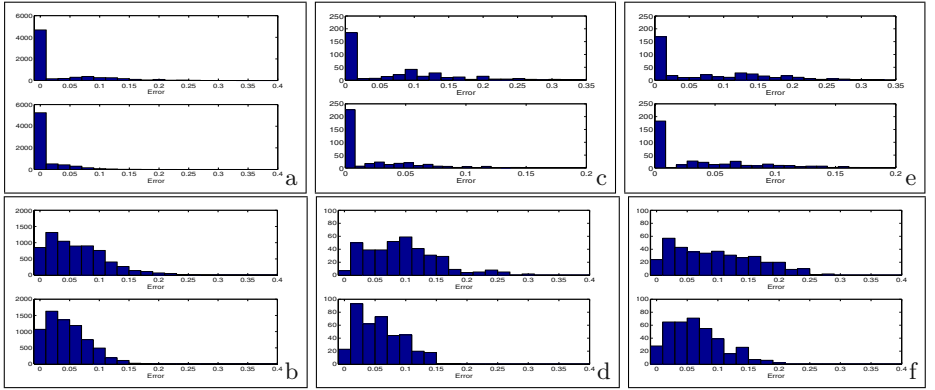
The first algorithm is probabilistic hierarchical clustering (PHC) [4]. It consists of a sequential binary Bayes classifier, which at each step evaluates a pair of video segments and decides on merging them into the same scene according to Gaussian mixture models of intra- and inter-scene visual similarity, scene duration, and temporal adjacency. The merging order and the merging criterion are based on the evaluation of a posterior odds ratio. The algorithm implicitly performs model selection. Standard visual features for each shot are extracted from key-frames (color histograms). Additionally, temporal features exploit the fact that distant shots along the temporal axis are less likely to belong to the same scene.

The second method uses spectral clustering (SC) [7], which has been shown to be effective in a variety of segmentation tasks. The algorithm first constructs a pair-wise key-frame similarity matrix, for which similarity is defined in both visual and temporal terms. After matrix pre-processing, its spectrum (eigenvectors) is computed. Then, the  $\mathcal{K}$  largest eigenvectors are stacked in columns in a new matrix, and the rows of this new matrix are normalized. Each row of this matrix constitutes a feature associated to each key-frame in the video. The rows of such matrix are then clustered using  $K$ -means (with  $\mathcal{K}$  clusters), and all key-frames are labeled accordingly. Shots are finally clustered based on their key-frame labels by using a majority vote rule. Model selection is performed automatically using the eigengap, a measure often used in matrix perturbation and spectral graph theories. The algorithm uses the same visual and temporal features as PHC, adapted to the specific formulation.



## 6.2 Results and Discussion

Figs. 3(c-f) show the error distributions when comparing the scenes found by people and the two automatic algorithms. The means for all measures are shown in Table 1. Comparing the results to those in Figs. 3(a-b), the errors for the automatic algorithms are higher than the errors among people. The degradation is more noticeable for the  $GCE$  and  $LCE$  measures, with a relative increase of more than 100% in the mean error for all cases. These results suggest that the automatic methods do not extract the scene structure as consistently as people do. In contrast, the relative variations in the correction factor are not so large. Overall, the automatic methods increase the error for  $GCE'$  and  $LCE'$ : 53.3% and 52.7% for  $GCE'$ , and 27.8% and 41.0% for  $LCE'$ , for PHC and SC, respectively.



**Fig. 3.** (a-b) Human scene structuring; (a) distributions of  $GCE$  (top) and  $LCE$  (bottom) for all pairs of video scene structures (same videos) in the database; (b) distributions of  $GCE'$  and  $LCE'$ . (c-d) PHC vs. human: (c)  $GCE$  (top) and  $LCE$  (bottom); (d)  $GCE'$  and  $LCE'$ . (e-f) SC vs. human: (e)  $GCE$  and  $LCE$ ; (f)  $GCE'$  and  $LCE'$ .

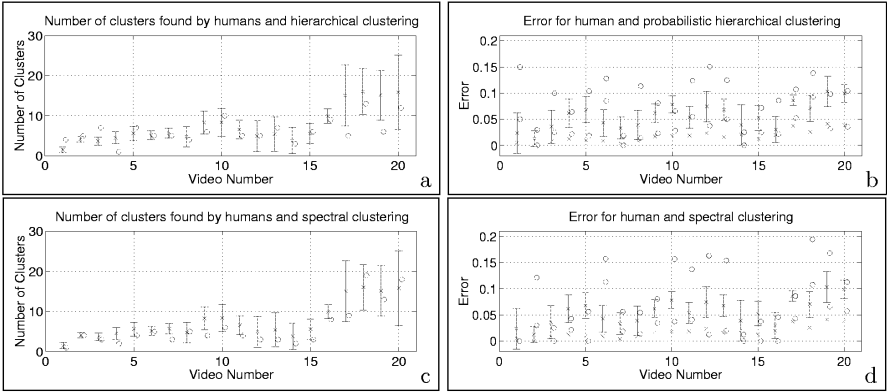
The results of our protocol appear in Table 2. Again, the error by automatic algorithms vs. people is higher than the errors among people, and the performance for both PHC and SC is quite similar. We used a two-tailed Wilcoxon signed-rank test analysis to detect significant differences between the two automatic algorithms for the min, med, and max performance over all videos. The obtained p-values are 0.658, 0.970, and 0.881, respectively, so the difference in performance is not statistically significant. In contrast, the tests comparing human vs. SC produced p-values of 0.147, 0.030, and 0.004, respectively, which indicates that the difference in performance for the min is only significant at  $p < 0.15$  level, but the differences for med and max are significant at  $p < 0.05$  and  $p < 0.005$  levels, respectively. Similar results are obtained when comparing human vs. PHC with the Wilcoxon test. Examples of human- and computer-generated video scene structures can be seen at [www.idiap.ch/~gatica/homevideoassess.html](http://www.idiap.ch/~gatica/homevideoassess.html). Note

that, although PHC and SC do not perform significantly different under this similarity measure, previous work using a different measure had favored SC [7].

**Table 1.** Error means. Human vs. human and automatic vs. human.

Case	$GCE$	$LCE$	$C$	$GCE'$	$LCE'$
human/human	0.0321	0.0119	0.2416	0.0635	0.0463
PHC/human	0.0656	0.0216	0.2725	0.0966	0.0592
SC/human	0.0740	0.0377	0.2214	0.0962	0.0653

Fig. 4 displays the results for each video for the two automatic algorithms. Figs. 4(a) and 4(c) show the number of detected scenes (red circles), and compare them to the mean number of scenes in the GT (blue crosses). The blue bar denotes the std in the GT. For both algorithms, the detected number of scenes matches well the GT, although somewhat underestimated. The number of scenes estimated by PHC (resp. SC) remain within one std of the mean human performance in 15 (resp. 17) of the 20 videos; in addition, in 14 (resp. 18) cases, the automatic method detected exactly the same number of scenes as at least one person did in the GT. These numbers are in agreement with the column for  $C$  in Table 1.



**Fig. 4.** Automatic (circles) vs. human (crosses) scene structuring. Top row: PHC. Bottom row: SC. (a-c) Number of detected scenes. (b-d)  $GCE'$  error. The bar is the spread of human performance (see text for details).

Figs. 4(b) and 4(d) show  $GCE'$  compared to the average of human performance. The circles denote the measures obtained with PHC/SC, the crosses denote human performance. Distinct colors represent different measures (minimum in red, median in blue, maximum omitted for space reasons). The median performance of PHC (resp. SC) stays within or below one std of the median human performance (i.e., blue circles within or below blue bars) in 9 (resp. 12) videos.

**Table 2.** Error means over individual performance.

Case	$GCE'_{min}$	$GCE'_{med}$	$GCE'_{max}$
human/human	0.0168	0.0563	0.1436
PHC/human	0.0333	0.0941	0.1827
SC/human	0.0308	0.0932	0.1870

## 7 Conclusions

We presented a methodology to benchmark scene structuring algorithms in home videos, using human performance on the task as the baseline. The agreement measures, adapted from work on natural image segmentation, attempt to model two concepts in perceptual organization. On a small but diverse data set, our experiments suggest that there exists human agreement in terms of organization of video scenes, but that there is a considerable variation w.r.t. scene granularity, which seems to depend on the visual content complexity. The comparison of two techniques with our methodology suggested that both performed similarly well, but still not as well as people. A comprehensive study that compares other agreement measures [9,4] and structuring algorithms remains as a future goal.

**Acknowledgements.** We thank the Swiss NCCR on Interactive Multimodal Information Management IM2 for support, and Eastman Kodak for the home video database.

## References

1. K. Boyer and S. Sarkar, "Perceptual Organization in Computer Vision: Status, Challenges, and Potential," *CVIU*, Vol. 76, No. 1, Oct. 1999.
2. S. Edelman, "Visual Perception", in *Encyclopedia of Artificial Intelligence*, 2:1655-1663, S. Shapiro, (ed)., Wiley, 1992.
3. S. Eickeler and S. Muller, "Content-based Indexing of TV News Using HMMs," in *Proc. IEEE ICASSP*, Phoenix, 1999.
4. D. Gatica-Perez, A. Loui, and M.-T. Sun, "Finding Structure in Home Videos by Probabilistic Hierarchical Clustering," in *IEEE T-CSVT*, Vol. 13, No. 6, Jun. 2003.
5. D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A Database of Human Segmented Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," in *Proc. IEEE ICCV*, Vancouver, Jul. 2001.
6. J.R. Kender and B.L. Yeo, "On the Structure and Analysis of Home Videos," in *Proc. ACCV*, Taipei, 2000.
7. J.-M. Odobez, D. Gatica-Perez, and M. Guillemot, "Spectral Structuring of Home Videos," in *Proc. CIVR*, Urbana, Jul. 2003.
8. K. Rodden, W. Basalaj, D. Sinclair, and K. Wood "Does Organization by Similarity Assist Image Browsing?," in *Proc. SIGCHI 2001*, Seattle, Apr. 2001.
9. D.M. Squire and T. Pun, "Assessing Agreement Between Human and Machine Clusterings of Image Databases," *Pattern Rec.*, Vol. 31, No. 12, 1998.
10. J. Vendrig and M. Worring, "Systematic Evaluation of Logical Story Unit Segmentation," *IEEE Trans. on Multimedia*, Vol. 4, No. 4, Dec. 2002.

# A Visual Model Approach for Parsing Colonoscopy Videos

Yu Cao<sup>1</sup>, Wallapak Tavanapong<sup>1</sup>, Dalei Li<sup>1</sup>, JungHwan Oh<sup>2</sup>,  
Piet C. de Groen<sup>3</sup>, and Johnny Wong<sup>1</sup>

<sup>1</sup> Department of Computer Science, Iowa State University  
Ames, IA 50011-1040, USA  
{[impact.isu@cs.iastate.edu](mailto:impact.isu@cs.iastate.edu)}

<sup>2</sup> Department of Computer Science and Engineering  
University of Texas at Arlington, Arlington, TX 76019-0015, USA  
[oh@cse.uta.edu](mailto:oh@cse.uta.edu)

<sup>3</sup> Mayo Medical School  
Mayo Clinic and Foundation, Rochester, MN 55905, USA

**Abstract.** Colonoscopy is an important screening procedure for colorectal cancer. During this procedure, the endoscopist visually inspects the colon. Currently, there is no content-based analysis and retrieval system that automatically analyzes videos captured from colonoscopic procedures and provides a user-friendly and efficient access to important content. Such a system will be valuable as an educational resource for endoscopic research, a platform to assess procedural skills for endoscopists, and a platform for mining for unknown abnormality patterns that may lead to colorectal cancer. The first necessary step for the analysis is parsing for semantic units. In this paper, we propose a new visual model approach that employs visual features extracted directly from compressed videos together with audio analysis to discover important semantic units called scenes. Our experimental results show average precision and recall of 93% and 85%, respectively.

## 1 Introduction

Colorectal cancer is the second leading cause of all cancer deaths behind lung cancer in the United States [1]. As the name implies, colorectal cancers are malignant tumors that develop in the colon and rectum. The survival rate is higher if the cancer is found and treated early before metastasis to lymph nodes or other organs occurs. Colonoscopy allows for inspection of the entire colon and provides the ability to perform a number of therapeutic operations during a single procedure. During a colonoscopic procedure, a tiny video camera at the tip of the endoscope generates a video signal of the internal mucosa of the colon. The video data are displayed on a monitor for real-time analysis by the endoscopist. The video data are not typically captured for post review or analysis. We call videos captured during colonoscopic procedures *colonoscopy videos*.

To the best of our knowledge, there is no content-based retrieval system that automatically analyzes colonoscopy videos and provides a user-friendly and

efficient access to important content. Such a system will be valuable as an important educational resource for endoscopic research, a platform to assess procedural skills for endoscopists, and a platform for mining for unknown abnormality patterns that may lead to colorectal cancer. Colonoscopy videos have unique characteristics, rendering known definitions of semantic units such as shots and scenes inapplicable. New definitions are required. Colonoscopy videos contain many blurry frames due to frequent shifts of camera focus while moving along the colon. Current endoscopes are equipped with a single, wide-angle lens that cannot be focused. Sharpness, brightness and contrast of the image therefore are optimized using the endoscopist's skills.

We have recently developed a new framework for parsing colonoscopy videos [2]. The framework includes a new scene definition and a novel audio-based scene segmentation algorithm. In this paper, we introduce a new visual model that captures a special kind of a cut-like and fade-like pattern appearing frequently around scene boundaries of colonoscopy videos. The pattern corresponds to the endoscopist's action of searching for the next anatomic landmark in the colon. Our new segmentation algorithm employs visual analysis in compressed domain based on the visual model together with our audio-based scene segmentation.

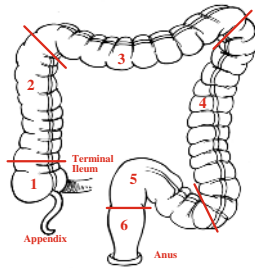
The remainder of this paper is organized as follows. Section 2 briefly provides background on our audio-based scene segmentation and related work. We present our new scene segmentation algorithm in Section 3. Experimental results are discussed in Section 4. Finally, we offer our concluding remarks in Section 5.

## 2 Background and Related Work

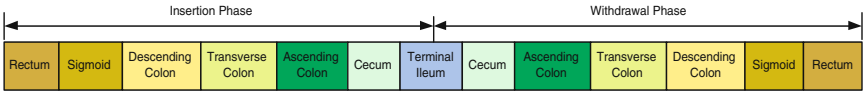
We briefly summarize our work on audio-based scene segmentation for colonoscopy videos. To the best of our knowledge, no visual analysis for scene segmentation on colonoscopy videos has been investigated. Our observation on these videos suggests that typical scene segmentation techniques using visual properties developed for produced videos (e.g., sports, news clips) are not applicable. However, hard cut and fade detection techniques are relevant and are summarized here.

### 2.1 Audio-Based Scene Segmentation for Colonoscopy Videos

The colon is a hollow, muscular tube about 6 feet long as illustrated in Fig. 1. A normal colon consists of six important parts: cecum with appendix, ascending colon, transverse colon, descending colon, sigmoid and rectum. A colonoscopic procedure consists of two phases: insertion phase and withdrawal phase. During the insertion phase, the endoscopist rapidly advances the tip of the endoscope to the most proximal location possible (cecum or terminal ileum). Careful mucosal examination, biopsy and therapeutic operations are typically performed during the withdrawal phase when the endoscope is gradually withdrawn.



**Fig. 1.** The colon endoscopic segments: 1-cecum, 2-ascending colon, 3-transverse colon, 4-descending colon, 5-sigmoid, 6-rectum



**Fig. 2.** Scenes of a colonoscopy video

A *scene* is defined as a segment of visual and audio data that correspond to an important part of the colon. Since a typical colon has six different parts and as the terminal ileum is also reachable during endoscopy, in a complete colonoscopy, a total of thirteen scenes are expected (see Fig. 2).

Our colonoscopy videos include the video signal from the endoscopy unit and the endoscopist’s dictation when the tip of the endoscope is moving from one colonic segment into the next. The endoscopist speaks pre-defined terms during the colonoscopic procedure to indicate the current position of the video camera. Examples of these terms are “Entering rectum”, “Leaving rectum, entering sigmoid”. “Leaving sigmoid, entering descending colon”, etc. In addition, the endoscopist may say terms indicating abnormality such as polyps and cancer. No patient identifiable information is included in the videos. Our audio-based scene segmentation algorithm works as follows.

First, audio frames are classified into four types: *Silence*, *Marker*, *Speech*, and *Background*. *Marker* indicates a special sound pattern of the change in the microphone status. To determine the type of each frame, a threshold-based algorithm using Short-Time Energy, Zero-Crossing Rate, Pitch, and Spectrum Flux is applied. Only the audio frames of the speech type are given to speech recognition software that outputs the corresponding text transcript.

We identify the name of each scene and associated boundaries as follows. We classify recognized words in the transcript into six categories: *Location*, *Action*, *Position*, *Abnormal*, *Error*, and *Unused*. The location category includes words describing important anatomic landmarks of the colon such as cecum and terminal ileum. The action category includes words indicating the action of the endoscopist such as “entering”. The position category has words indicating the camera position such as “begin” and “end”. The abnormal category has terms indicating abnormality such as “polyp” and “cancer”. The unused category in-

cludes non-communicative words such as “uh”. Speech segments that cannot be recognized are assigned to the error category. A finite state automaton that recognizes the regular expression:  $(Action \vee Position)^* \cdot Location$  is used to recognize words spoken at scene boundaries.

Based on the transcript and the timestamp of each speech segment, we obtain the scene boundaries as follows. Starting from the first speech segment, we locate the nearest speech segment with the same word in the location category (e.g., “rectum” in “entering rectum” and in “leaving rectum, entering sigmoid”). The starting time of the former speech segment and the ending time of the latter speech segment indicate the scene boundaries.

## 2.2 Hard Cut and Fade Detection Techniques for Produced Videos

**Hard Cut Detection:** A hard cut is a direct concatenation of two shots, which indicates a temporal, abrupt visual discontinuity in a video. Existing hard cut detection techniques detect significant changes in either intensity/color histograms [3,4,5] or edge pixels [6] or motions [7] between consecutive frames.

**Fade Detection:** A production model of a fade sequence  $S(x, y, t)$  of duration  $T$  is defined as the scaling of pixel intensities/color of a video sequence  $S_1(x, y, t)$  by a temporally monotone scaling function  $f(t)$  as in Equation (1)[8].

$$S(x, y, t) = f(t) \times S_1(x, y, t), t \in [0, T] \quad (1)$$

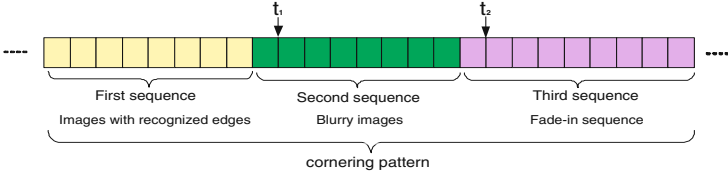
For a fade-in sequence,  $f(0) = 0$  and  $f(T) = 1$  while  $f(0) = 1$  and  $f(T) = 0$  for a fade-out sequence. Typically,  $f(t)$  is a linear function. It was observed that a fade detector based on edge changes does not perform as well as a fade detector based on changes in standard deviations of pixel intensities [9].

## 3 New Scene Segmentation Approach

Our proposed scene segmentation consists of two steps. First, we apply our audio-based scene segmentation algorithm discussed in Section 2.1. However, some scenes may not be detected because the endoscopist’s speech is not recognized by the speech recognition software. Domain knowledge about the scenes in a typical colonoscopy video is helpful in identifying the names of the missing scenes, but is unable to identify the boundaries of these scenes. We apply visual analysis based on our new visual model to determine the missing scene boundaries.

### 3.1 Visual Model for Scene Segmentation

Based on our observations and consultations with our endoscopist, we observe a specific pattern appearing around 60% of scene boundaries in colonoscopy videos. We call this pattern the *cornering pattern* as it corresponds to the endoscopist’s action of steering the endoscope around the cornering parts of the colon (i.e., cecum and terminal ileum, ascending and transverse colons, transverse and descending colons, and descending and sigmoid colons). The cornering



**Fig. 3.** Cornering pattern around a scene boundary

pattern consists of three sequences of images (see Fig. 3). The first sequence is composed of images with recognized edges. The second sequence has all blurry images—images with unclear edges. The transition between these two sequences is quite abrupt like a hard cut in produced videos. The third image sequence is like a fade-in sequence with a gradual increase in pixel intensities/color and edges. This sequence happens as the endoscopist starts to recognize some part of an anatomic landmark and gradually adjusts the camera position to make the image clearer. Existing production models [8,10] cannot capture the cornering pattern. Hence, we propose a new visual model for this pattern. Let  $S_1(x, y, t)$ ,  $S_2(x, y, t)$ , and  $S_3(x, y, t)$  represent the first, the second, and the third image sequences, respectively. The spatial dimension is represented by  $x$  and  $y$  and the temporal dimension is represented by  $t$ . Thus, the cornering pattern  $S(x, y, t)$  is defined in Equation(2).

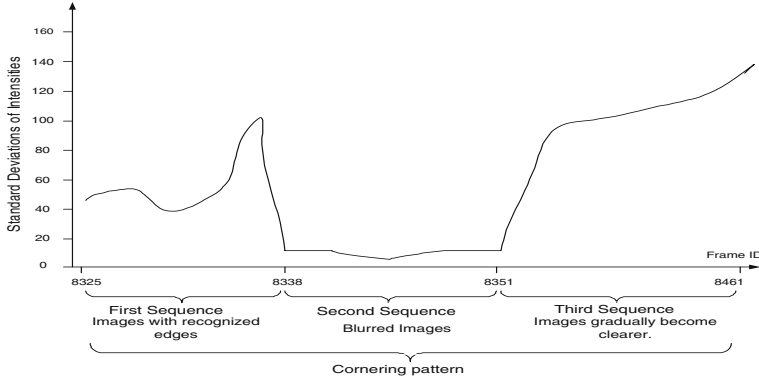
$$\begin{aligned}
 S(x, y, t) = & (1 - H(t - t_1)) \times S_1(x, y, t) + \\
 & H(t - t_1) \times (1 - H(t - t_2)) \times S_2(x, y, t) + \\
 & H(t - t_2) \times f(t - t_2) \times S_3(x, y, t)
 \end{aligned} \tag{2}$$

where  $t_1$  denotes the timestamp of the first frame after the first sequence and  $t_2$  is the timestamp of the first frame after the second sequence (see Fig 3).  $H(t)$  is a function that outputs 1 when  $t \geq 0$  and 0 otherwise. When  $t < 0$ ,  $f(t)$  produces zero; otherwise, the function is a temporally scaling function. This function is typically not a linear function as in the case of a production model for a typical fade sequence.

### 3.2 Feature Extraction and Analysis

Since our colonoscopy videos are already encoded in MPEG-2, we extract visual features directly from the compressed videos to reduce the segmentation time. We first obtain a DC-image from the Y-color plane (intensity) of each frame using the technique in [11]. A DC-image is a spatially reduced version of the original image. We compute the standard deviation of DCT coefficients in each DC-image. This is based on our observation that the distribution of the standard deviations of the DC images in the cornering pattern often follows the pattern in Fig. 4. That is, the standard deviation of each DC-image in the second sequence is generally small and smaller than those of the frames in the other two sequences. We call the second sequence *monotone sequence*. We observe that the standard



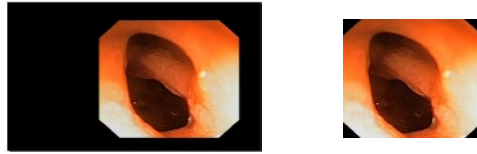


**Fig. 4.** Pattern of standard deviations of DC images in the cornering pattern

deviations of the frames in the fade-in sequence can be modeled using a curve fitting method. We choose a linear regression model to describe the standard deviations of the frames in the third sequence by one or more linear functions. The challenge is to find the ending frame of each linear curve automatically. Hence, the scaling function  $f(t)$  in Equation(2) may be a combination of one or more linear functions.

### 3.3 Scene Boundary Detection Algorithm

**Step 1: Preprocessing:** Since more than 99% of the scene boundaries fall in the speech segments, we restrict visual analysis on the video segments corresponding to the endoscopist's speech segments excluding those that contain the keyword in the abnormal category. This is because the terms in this category are very specific and irrelevant to scene boundaries. Next, we apply the filter that removes the black area (the area with DC coefficients below a threshold) surrounding the useful portion of the image (see Fig. 5).



**Fig. 5.** Captured image and image after removal of the black surrounding region

**Step 2: Detection of a monotone sequence:** A sequence of consecutive frames is declared as a monotone sequence if it has at least a pre-defined minimum number of consecutive frames with the standard deviation of each of these frames below a *monotone threshold*.

**Step 3: Hard Cut Detection:** To check a discontinuity between the first sequence and the monotone sequence, we use a sliding-window of size  $2w + 1$

consecutive frames. We first position the center of the sliding window at the frame immediately before the first frame in the monotone sequence. We derive a sequence of bin-wise histogram differences between DC-images of two consecutive frames in the window. We declare a hard cut at the center of the sliding-window if the histogram difference of the two consecutive frames at the center is the largest within the window, and the ratio between the largest difference and the second largest difference in the window is larger than a predefined *hard-cut ratio*. If a cut is not found, we slide the window away from the monotone sequence by one frame. The same process is repeated until a cut is found or a given number of frames before the monotone sequence have been checked. In the latter case, no hard cut is detected.

**Step 4: Detection of a fade-in sequence:** We check whether two linear curves fit well with the standard deviations of the coefficients of DC-images after the monotone sequence using the algorithm in Fig. 6.

---

```

/* Let  $\sigma_i$  be the standard deviation of the coefficients in the DC-image of frame  $i$  */
 $e$  := frame ID of the last frame in the monotone sequence
 $i$  := 0;  $c$  := 0;
repeat
   $n$  := 2; /* consider the ending frame of the previous sequence and  $n$ 
            subsequent frames */
  repeat /* correlation coefficient value is in the range  $[0, 1]$  */
     $r_1^2$  is a correlation coefficient of  $\sigma_e, \dots, \sigma_{e+n}$ 
     $r_2^2$  is a correlation coefficient of  $\sigma_e, \dots, \sigma_{e+n+1}$ 
     $n$  :=  $n + 1$ ;
  while  $r_1^2 - r_2^2 < (0.05 \cdot r_1^2)$  /* the change in correlation values is small */
  if  $r_1^2 > 0.8$  then  $c$  :=  $c + 1$ ; /* a linear curve fits well with the values */
   $i$  :=  $i + 1$ ;  $e$  :=  $e + n$ ;
while  $i < 2$ ;
if  $c = 2$ , a fade-in sequence is detected

```

---

**Fig. 6.** Fade-in sequence detector for a cornering pattern

**Step 5: Boundary Identification:** If both a monotone sequence and a fade-in sequence are detected, the scene boundary is declared at the first frame after the ending frame of the fade-in sequence. However, if a hard cut and a monotone sequence are detected without the fade-in sequence, we declare the scene boundary at the hard-cut location.

## 4 Performance Study

We first obtain appropriate parameter values for the proposed visual analysis technique using a training set of ten colonoscopy videos. Like scene boundaries for produced videos, scene boundaries for colonoscopy videos are also subjective. Since the movement of the camera is slow, the anatomic landmark used for scene boundary identification by the endoscopist appears in several frames. Therefore,

if the software detected boundary is within two seconds from the boundary determined by the endoscopist, we treat the detected boundary as a correct boundary. A detected scene is considered correct if both of the detected boundaries of the scene are correct. We measure the following performance metrics.

$$\text{Recall} = \frac{\text{Number of correctly detected scenes}}{\text{Number of actual scenes}}$$

$$\text{Precision} = \frac{\text{Number of correctly detected scenes}}{\text{Number of detected scenes}}$$

Table 1 shows the performance comparison of the fade-in detector using one linear curve (“Model 1”), two linear curves (“Model 2”), and three linear curves (“Model 3”). The fade-in detector with two linear curves produces the highest precision and recall and is used in the subsequent performance comparison.

**Table 1.** Effect of fade detection models on the training set

	Model 1	Model 2	Model 3
<i>Precision</i>	0.91	0.94	0.86
<i>Recall</i>	0.62	0.78	0.71

**Table 2.** Precision and recall of three scene segmentation algorithms

<i>ID</i>	<i>Length</i> (min)	<i>Precision</i>			<i>Recall</i>			<i>Time (sec.)</i>		
		AO	AV-C	AV-U	AO	AV-C	AV-U	AV-C	AV-U	$\frac{AV-C}{AV-U}$
001	18:26	0.89	0.90	0.91	0.62	0.69	0.77	2597	7150	0.36
007	25:08	0.90	0.91	0.91	0.69	0.77	0.77	3600	10588	0.34
009	37:22	0.91	0.85	0.85	0.77	0.85	0.85	5310	13973	0.38
010	34:24	1.00	1.00	1.00	0.85	0.85	0.85	4905	14420	0.34
014	36:33	0.91	0.77	0.85	0.77	0.77	0.85	5199	14442	0.36
015	23:00	1.00	1.00	1.00	1.00	1.00	1.00	3317	8965	0.37
017	21:14	0.90	0.90	0.90	0.69	0.69	0.69	3029	8413	0.36
019	24:05	0.90	0.92	1.00	0.69	0.85	0.92	3466	9903	0.35
020	13:07	1.00	1.00	1.00	1.00	1.00	1.00	1800	4800	0.38
047	28:29	0.90	0.82	0.82	0.69	0.69	0.69	4037	11214	0.37
062	30:34	0.83	0.77	0.77	0.77	0.77	0.77	4328	11697	0.37
133	33:02	1.00	1.00	1.00	0.85	1.00	1.00	4762	12806	0.37
148	24:28	1.00	1.00	1.00	0.92	0.92	0.92	3460	9582	0.36
152	11:55	1.00	1.00	1.00	0.92	1.00	1.00	1587	4376	0.36
163	19:34	1.00	1.00	1.00	0.92	1.00	1.00	2742	7374	0.37
177	21:29	0.89	0.89	0.89	0.62	0.62	0.62	3031	8156	0.37
179	29:15	1.00	1.00	1.00	0.69	0.77	0.77	4184	11252	0.37
185	21:34	1.00	1.00	1.00	0.92	0.92	0.92	3049	8168	0.37
190	27:07	0.91	0.92	1.00	0.77	0.92	1.00	3896	10477	0.37
197	14:54	1.00	1.00	1.00	0.85	0.85	0.85	2020	5437	0.37
Average	24:44	0.95	0.93	0.95	0.81	0.85	0.86	3516	9560	0.36

## 4.1 Performance Comparison

Given the best parameter values obtained via experiments with the training set, Table 2 shows the performance comparison among our audio-based segmentation (AO), our model approach using features in compressed domain (AV-C), and our model approach using features derived from pixel intensities of uncompressed videos (AV-U) on twenty colonoscopy videos not included in the training set. Both model-based approaches outperform the audio-based technique in terms of recall. AV-C and AV-U found 11 and 15 correct scenes missed by AO, respectively. AV-U performs slightly better than AV-C because AV-U can better detect the boundaries of the terminal ileum scene. On average, AV-C takes only about a third of the time taken by AV-U to segment a video on the same machine. Hence, a hybrid approach using AV-U for detecting boundaries of the terminal ileum scene and AV-C for the other scenes should give the best recall and precision with the segmentation time in between that of AV-C and AV-U.

## 5 Conclusion Remarks

We have presented a scene segmentation technique for videos generated from colonoscopic procedures. The technique employs audio analysis and a new visual analysis method based on our visual model for colonoscopy videos. Experiments on real colonoscopy videos show average precision and recall of 93% and 85%, respectively. Future works include benign and malignant lesions detection, video annotation, video summarization, and video browsing and retrieval.

## References

1. Greenlee, R., Murry, T., Bolden, S., Wingo, P.A.: Cancer statistics. *CA Cancer J Clin* **50** (2000) 7–33
2. Cao, Y., Tavanapong, W., Kim, K.H., Wong, J., Oh, J.H., de Groen, P.C.: A framework for parsing colonoscopy videos for semantic units. In: To appear in *Proc. of Int'l Conf. on Multimedia and Expo, Taipei, Taiwan* (2004)
3. U.Gargi, Kasturi, R., S.H.Strayer: Performance characterization of video-shot-change detection methods. *IEEE Transaction on Circuits and Systems for Video Technology* **10** (2000) 1–13
4. Yusoff, Y., Kittler, J.: Video shot cut detection using adaptive thresholding. In: *Proc. of the British Machine Vision Conference, Bristol, UK* (2000)
5. Naphade, M.R., Mehrotra, R., Ferman, A.M., Warnick, J., Huang, T.S., Tekalp, A.M.: A high-performance shot boundary detection algorithm using multiple cues. In: *Proc. of the IEEE Int'l Conf. on Image Processing, Chicago, Illinois, USA* (1998) 884 – 887
6. R.Zabih, J.Miller, K.: A feature-based algorithm for detecting and classification production effects. *Multimedia Systems* **7** (1999) 119–128
7. Hanjalic, A., Zhang, H.J.: Optimal shot boundary detection based on robust statistical models. In: *Proc. of the IEEE Int'l Conf. Multimedia Computing and Systems, Florence, Italy* (1999)

8. Hampapur, A., Jain, R., Weymouth, T.: Production model based digital video segmentation. *Multimedia Tools and Applications* **1** (1995) 9–46
9. Lienhart, R.: Comparison of automatic shot boundary detection algorithms. In: *Proc. of SPIE Storage and Retrieval for Still Image and Video Databases VII*. Volume 3972. (1999) 290 – 301
10. Truong, B.T., Dorai, C., Venkatesh, S.: New enhancements to cut, fade, and dissolve detection processes in video segmentation. In: *Proc. of ACM Multimedia*, Los Angeles, CA, USA (2000) 219–227
11. Yeo, B.L., Liu, B.: Rapid scene analysis on compressed video. *IEEE Transactions on Circuits and Systems for Video Technology* **5** (1995) 533–544

# Video Summarization and Retrieval System Using Face Recognition and MPEG-7 Descriptors

Jae-Ho Lee and Whoi-Yul Kim

Image Engineering Laboratory, Division of Electrical and Computer Engineering,  
Hanyang University, Seoul, Korea  
jhlee@vision.hanyang.ac.kr, wykim@hanyang.ac.kr  
<http://vision.hanyang.ac.kr>

**Abstract.** In this paper, we introduce an automatic video summarizing and indexing tool for a personal video recorder. The tool utilizes MPEG-7 visual descriptors to generate a video index for summary. The resulting index generates not only a preview of a movie but also allows non-linear access with thumbnails. In addition, the index supports the searching of shot similar to a desired one within saved video sequences. Moreover, face recognition technique is utilized to personal based video summarization and indexing in stored video data.

## 1 Introduction

The popularization of digital broadcasting forces us to come into contact with a large amount of video data. The sheer amount of data is becoming increasingly difficult to handle on conventional home electronics. Utilization of the MPEG-7 is a reasonable approach to describe and manage multimedia data [1]. To this end, there has been some research done on the use of MPEG-7 in broadcasting content applications. A. Yamada et al. have built a visual program navigation system that uses an MPEG-7 color layout descriptor [2]. N. Fatemi and O.A. Khaled designed a retrieval application using a rich news description model based on the MPEG-7 standard [3], and T. Walker proposed a system for content-based navigation of television programs based on the MPEG-7 [4]. The system that Walker presented uses standard MPEG-7 description schemes (DS) to describe television programs, but is limited to news programs. T. Sikora applied the MPEG-7 descriptor for the management of multimedia databases [5]. Also, A. Divakaran et al. presented a video summarization technique using cumulative motion activity based on compressed domain features extracted from motion vectors [6]. The MPEG-7 standard is composed of seven main parts [7]. The visual component is categorized by basic structures such as color, texture, shape, motion, localization and face recognition. A detailed explanation of all visual descriptors can be found in the standard document [8].

In this paper, we introduce a video summarizing system that generates summarized video efficiently. The summary information generated by the presented tool provides users an overview of video content, and guides them visually to move to a desired position quickly in a video. Also, the tool makes it possible to find shots similar to a

queried one or face, which is useful for editing different video streams into a new one. Here, MPEG-7 visual descriptors are only used to segment a video into shots, to summarize a video, and to retrieve a scene of interest. This work is extension of our previous research [18].

## 2 Implementation of Summarizing System

The functions of a developed system consist of mainly four parts:

1. Generation of an overview of video contents; to find a desired position in video data.
2. Query by example or face; to find similar shots to a queried one in a large amount of video data.
3. Nonlinear editing; to support simple editing based on the summarized video.
4. Actor based indexing; to find scenes which contains queried person in a video.

Each image in the summary index becomes a representative frame of the cluster which is composed of a number of shot segments. Users can find similar shots to his or her favorite shot by querying with an example in the index. Also, a new video stream can be easily generated by moving shot segments from the summary index. All of these operations have been designed simply, so they can be executed with a home electronics interface. The implemented functions in the developed tool are described in the followed sub sections.

### 2.1 Video Summarization

With our tool, a video summary is generated in three ways: semantic-based, content-based, and face-based summarization. Semantic summarization is for videos that have stories such as dramas or sitcoms, while content-based summarization is for videos that do not contain stories such as sports videos. In both cases, a video stream is segmented into a set of shots as a preprocessing step by detecting scene changes. The adaptive threshold and gradual transition detection methods are applied in this step [9][10]. In this stage, three MPEG-7 color descriptors: color layout, dominant color, and color structure are computed and saved as the features to be utilized in the summarization and retrieval process.

Abrupt scene change detection is done simply by computing the distance between two sets of features extracted from adjacent frames. Since the number of abrupt scene changes is highly dependent upon the threshold value, an adaptive method can be used using the average and deviation of the local duration [9]. To detect a gradual change, the distance value between the current frame and the ones at the  $k$  frame is computed by Bescos's plateau method [10]. To detect the exact location of the plateau form, metrics such as symmetry, slope fall (distance decreasing on rising phase, or vice versa), maximum of distance and distance difference when scene change occurs are used. Values of 10, 20, 30, and 40 were chosen as  $k$  for the various durations of gradual changes. The Color Layout Descriptor (CLD) was used as the feature of distance to detect these abrupt and gradual scene changes.

For key-frame extraction of the segmented video shots, some approaches have been proposed recently. H. Chang, S. Sull and S. Lee presented a method to measure the performance of key-frames [11]. Although this method is not infallible, it is certainly applicable to the results presented in this submission. It would enable them to compare the results with other techniques. Also, A. Hanjalic and H. Zhang provided a much more thorough review of the key-frame extraction techniques as well as perhaps another way to assess the performance of a key-frame extraction scheme using cluster-validity analysis [12]. In this paper, the oldest attempt to automate the key-frame extraction was adapted [13], because it chooses as a key frame the frame appearing after each detected shot boundary, and this method is appropriate for a PVR which receives video data via a broadcasting system.

After the scene change detection step, a video summarization process follows. To perform a semantic summarization, the segmented shots are clustered to compose story units. To obtain content-based summarization, clustering is applied while considering the duration of each shot segment. In both, the distance between shots is measured using a MPEG-7 color layout descriptor and a color structure descriptor.

By comparing key frames from the scene change detection step, the process of shot clustering is followed by a modified time-constrained method. Time-constrained clustering is based on the observation that similar shot segments separated in time have a high probability to be located in other story units. As a result, remote shots are not in the same story unit using a time windowing method, even though the shots have similar features. A hierarchical clustering method merges shot segments that have similar features and neighbor each other in the time domain into the identical cluster. The time window comparing regions is fixed as 3000 seconds. Yeung proposed a Scene Transition Graph (STG) to generate story units from clustering results [14]. Each node of the STG is a cluster, and links between clusters are generated when there are adjacent shot segments. To separate story units, the following observation was proposed: (1) shots in identical units interact with each other (2) there is no interaction with shots in the other story units except as one transition between units. With this observation, cut edge, which is one directional path between units, was estimated. In our system, we selected a simple numbering method to detect the transition point. The pseudo code of the method is shown below:

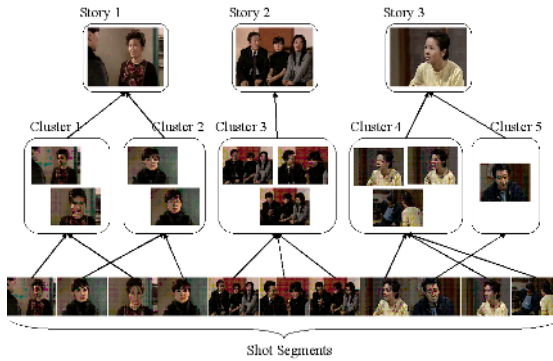
```

Story0 ← Cluster0, lastCID = 0, j=0
for i=0 ~ Number of Shots
    if(CID of shoti > lastCID)
    {
        j++
        Storyj ← CID of shoti
        lastCID = CID of shoti
    }
    else if(CID of shoti < CID of shoti-1)
    {
        merge Storyk+1, Storyk+2, ... , Storyj
        where Storyk contains the cluster which shoti is be-
        longed to
    }

```



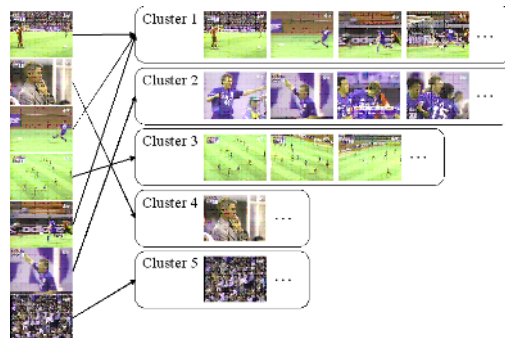
If there is a new Cluster ID while checking the Cluster ID of continuous shots, the shot is regarded as a new story unit. However, when an interaction is detected, all related story units are merged. For example, Semantic summarization has a hierarchical structure driven by considering the temporal locality and continuity of content in video. In this system, we assumed that the story is the top layer of the structure, and each story is organized by clusters. These clusters also have some key frames of the scene. Figure 1 describes the hierarchical structure of the semantic summarization.



**Fig. 1.** Hierarchical structure of semantic summary

The bottom images represent the key frame of each shot segment. All shot segments are included in the higher level of hierarchical structure which is called the cluster by comparing its features. And the story units hold one or more cluster as the highest level of the structures.

Content-based summarization focuses on the coincidence of content without temporal information. This method can be applied especially in sports videos. For example, in a soccer video, player scenes, goal scenes, and audience scenes can all be classified separately. Figure 2 shows the example of the content-based summarization result.



**Fig. 2.** Contents-based video summary

In the content-based summary, there is no hierarchical structure. The far left column depicts a video sequence arriving in the system. And the content-based summary results are displayed at the right side. Each cluster has key-frames of a similar feature, even if they are located separately in time.

Face recognition and detection techniques can also be utilized in summarization in video data. This summarization scheme can be utilized efficiently for user in accessing or retrieving specially in drama. Figure 3 shows the result of face based summarization in video data.

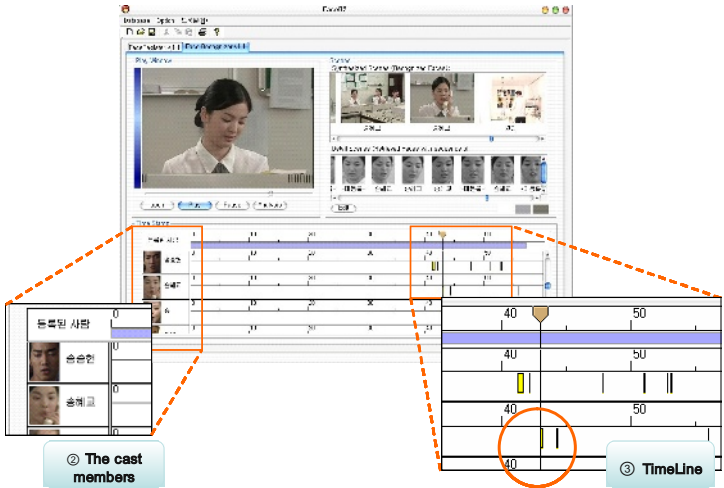


Fig. 3. The face based summarizing results.

The PCA/LDA face recognition method was adapted in recognition process [20]. And the Haar-like feature utilized in face detection stage to the pseudo real-time process [19].

## 2.2 Video Segment Editing

Video editing, which is based on a summarized index, is also supported with the developed system. In the editing procedure, the segmented shots can be removed or merged into a new video stream by the user. Users can generate a video stream that consists of their favorite shots using this editing tool. Each of these functions is designed to be used easily with the use of a remote control. The user chooses the shot segment, cluster, or story in the video indexes using a remote control. This only requires pushing the insert and move button on a remote control to edit. With this simple process, a user can make his or her favorite scenes easily.

### 2.3 Video Retrieval

A query of similar scenes can be achieved with the developed system. The MPEG-7 descriptors are used to find similar scenes in query by example methods. This function plays an important role by providing user convenience with quick searching of favorite scenes for editing or direct access. If a user orders a retrieval function by clicking the query image in one index, the similar key frames are retrieved along with all saved summary indexes.

## 3 Experimental Results

The MPEG-7 visual descriptors are utilized in the developed system to keep up with the further extension of home entertainment systems using internet connectivity. The detailed algorithms for extracting visual features and the similarity measurement between the features were referenced from the XM document and software in MPEG-7 [15][16]. To analyze the performance of the scene change detection with visual descriptors, two trailers and two music videos were selected as test video data. Usually, the detection of abrupt scene change is easier than gradual change. Therefore, we employed this test data because each includes a lot of gradual scene changes. In TABLE 1, the number of scene changes in each video is displayed. And, the results of detection are shown in TABLE 2. According to the results, the MPEG-7 color descriptors can be utilized as a feature for scene change detection.

**Table 1.** Information of videos in experiments

No	No. of frames	No. of abrupt changes	No. of gradual transitions
1	4005	43	29
2	3616	55	22
3	5552	0	34
4	6858	78	37

**Table 2.** The accuracy rate of scene change detection

No	Abrupt change		Gradual transition	
	Recall (%)	Precision (%)	Recall (%)	Precision (%)
1	93	98	81	88
2	94	98	88	65
3	-	-	87	77
4	95	97	81	89

The example of the scene change detection and key frame extraction of a movie is displayed in Figure 4. The key frame is selected as the first frame for each shot. From the experimental results, the average of cluster numbers decreased to 30 percent of

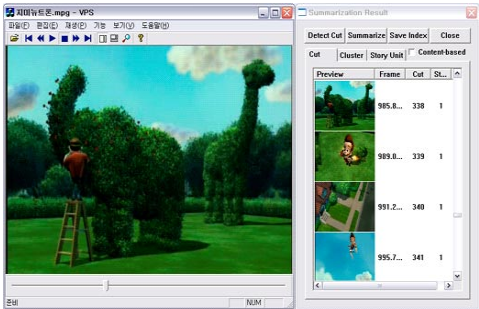


Fig. 4. Example result of an video summary

the total segmented shots. After the clustering procedure, all the clusters are gathered and classified into story units. The average of total number of story units is usually about 15 percent of the clusters. The generated shots, clusters and story units form a hierarchical structure can assist the consumer to view and access the video content easily. To analyze summarization efficiency, six types of video data were utilized. The summarized results are shown in TABLE 3.

Table 3. Experimental results on video summarization

Video	Duration	No. of shots	No. of clusters	No. of stories
Comedy show	45m 15s	757	279	34
Drama	19m 46s	174	47	6
Animation	27m 27s	630	267	14
Music show	83m 13s	1141	428	43
News	60m 36s	1066	712	66
Movie	39m 21s	472	167	12

The numbers of stories are dependent upon the characteristics of the video data. In the case of video data of a short duration shot such as news, music shows, and comedy shows, there are many shots, cluster, and stories in the summarized index results. On the other hand, drama, animation, and movie data have a small number of stories because it has long and similar video content, respectably.

The retrieval result of the queried key frame is presented in Figure 5(a). The color descriptors are also utilized. The compounding multi-feature can generate more accurate retrieval results. The detailed performance analysis of the compound color descriptors is described in [17].

The image on the top left is the query image in Figure 5(a). The similar frames are displayed in a window by retrieval result. A user can directly access and edit with this result. Figure 5(b) shows the example of the edited scene results with three individual dramas. The generated scene includes only one actress according to user choice. The new scene can also be saved for reproduction. On the top left hand side is the list of opened indexes, and on the right hand side it indicates the summarized index of the



**Fig. 5.** (a) The retrieval results in key frames, (b) The results of new video segment with three different video clips

selected files. The two bottom rows are edited scenes and generated scenes which have been selected by the user. The result of a newly generated scene is displayed in the middle.

## 4 Conclusions

The object of this research is to develop a summarizing system using only the visual descriptors in Part-3 without any human intervention or manual annotation. In this paper, we have introduced our video summarizing tool. The resulting tool enables users to access a video easily through a generated summarization index. In addition, the summarization index also supports other operations that can be helpful and interesting for users: querying a scene and editing a video stream. The MPEG-7 descriptors are used to obtain video summarization and to retrieve a queried scene. The proposed tool was devised to be operated inexpensively with a simple interface; therefore, it can be embedded in a PVR. Furthermore, the system can be extended for a video search engine with internet connectivity PVRs using the MPEG-7 technique.

## References

1. MPEG-7 Group: MPEG-7 Applications Document, ISO/IEC JTC1/SC29/WG11/N2462, Atlantic, October (1998)
2. A. Yamada, E. Kasutani, M. Ohta, K. Ochiai, and H. Matoba: Visual Program Navigation System based on Spatial Distribution of Color, IEEE Proc. of International Conference on Consumer Electronics, 13-5 (2000) 280-281

3. N. Fatemi and O.A. Khaled: Indexing and retrieval of TV news programs based on MPEG-7, IEEE Proc. of International Conference on Consumer Electronics, 20-6 (2001) 360-361
4. T. Walker: Content-based navigation of television programs using MPEG-7 description schemes, IEEE Proc. of International Conference on Consumer Electronics, 13-1 (2000) 272-273
5. T. Sikora: Visualization and Navigation in Image Database Applications based on MPEG-7 Descriptors, IEEE proc. of International Conference on Image Processing, vol. 3 (2001).583
6. A. Divakaran, R. Regunathan, and K.A.Peker: Video Summarization Using Descriptors of Motion Activity: A Motion Activity Based Approach to Key-Frame Extraction from Video Shots, Journal of Electronic Imaging, vol. 10, no. 4 (2001) 909-916
7. B. S. Manjunath et al.: Introduction to MPEG-7, John Wiley & Sons Ltd., West Sussex, England (2002).
8. ISO/IEC 15938-3, „Multimedia Content Description Interface - Part 3: Visual," version 1 (2001)
9. Y. Yusoff, W. Christmas, and J. Kittler: Video shot cut detection using adaptive thresholding, British Machine Vision Conference (2000).
10. J. Bescos, J.M. Menendez, G. Cisneros, J. Cabrera, and J.M. Martinez: A unified approach to gradual shot transition detection, IEEE Proc. of International Conference on Image Processing, vol. 3 (2000) 949 -952
11. H. Chang, S. Sull and S. Lee: Efficient Video Indexing Scheme for Content-Based Retrieval, IEEE Trans. on Circuits and Systems for Video Technology, vol.9 (1999) 1269-1279
12. A. Hanjalic and H. Zhang: An Integrated Scheme for Video Abstraction based on Unsupervised Cluster-Validity Analysis, IEEE Trans. on Circuits and Systems on Video Technology, vol. 9 (1999) 1280-1289
13. B. Shahraray and D. Gibbon: Automatic generation of pictorial transcripts of video programs, SPIE proc. of Multimedia Computing and Networking (1995) 512-518
14. M. Yeung and B.L. Yeo: Segmentation of video by clustering and graph analysis, Computer Vision and Image Understanding Journal, vol. 71, no. 1 (1998) 97-109
15. MPEG-7 Visual part of eXperimentation Model Version 10.0, ISO/IEC JTC1/SC29/WG11/N4063, Singapore (2001)
16. MPEG-7 Experimental Model Software, [http://www.lis.e-technik.tu-muenchen.de/research/bv/topics/mmdb/e\\_mpeg7.html](http://www.lis.e-technik.tu-muenchen.de/research/bv/topics/mmdb/e_mpeg7.html).
17. J.H. Lee, H.J. Kim, and W.Y. Kim: Video/Image Retrieval System (VIRS) based on MPEG-7, IEEE proc. of International Conference on Information Technology: Research and Education, submitted for publication (2003)
18. J.H. Lee, G.G. Lee, W.Y. Kim: Automatic Video Summarizing Tool using MPEG-7 Descriptors for Personal Video Recorder, IEEE Trans. On Consumer Electronics, Vol. 49, No. 3 (2003) 742-749
19. P. Viola and M.J. Jones: Robust real-time object detection, Technical Report Series, Compaq Cambridge research Laboratory, CRL 2001/01 (2001)
20. W.Y. Kim, J.H. Lee, H.S. Park, and H.J. Lee: PCA/LDA Face Recognition Descriptor using Pose Transformation, ISO/IEC JTC1/SC29/WG11 MPEG2002/M8934 (2002.)

# Automatic Generation of Personalized Digest Based on Context Flow and Distinctive Events

Hisashi Miyamori

Keihanna Human Info-Communication Research Center,  
National Institute of Information and Communications Technology (NICT),  
3-5, Hikari-dai, Seika-cho, Souraku-gun, Kyoto, 619-0289 Japan  
`miya@nict.go.jp`

**Abstract.** This paper proposes an method of automatically generating video digest that can dynamically change the scene component ratio in relation to context semantics and to individual distinctive events. We implemented a system of generating tennis digest, where three parameters given by the user, i.e. player being focused on, digest composition, and total duration, were reflected in the content. We asked several people with TV production experience to create correct digest data for reference, which should be generated when the same parameters are given. The reference data and the output of the system were compared to evaluate the effectiveness of the proposed method. The results revealed that the generated digests could convey the semantics of the original video reasonably well, and they demonstrated the performance and the validity of our approach.

## 1 Introduction

Recently, the amount of visual information available has been rapidly increasing across various fields. Video summarization is expected to become increasingly important, considering its capability of enabling information to be accessed more efficiently and important segments or highlights to be browsed from the entire content within a limited time.

The previous approaches to video summarization can be classified into two.

The first has mainly focused on automatically extracting low-level features from various media, such as color, texture, camera motion, facial characteristics, captions, sound classification, and TF-IDF for transcripts. It identifies important scenes by using a combination of these features and their transitions[1]-[4]. Many examples have been reported where it has been applied to video that has a comparatively simple structure, such as news videos. A common drawback has been that it becomes difficult to identify specific semantic content such as what happens in each individual scene, since the method is based on low-level features.

The second has involved several researches which have allowed considerable manual input of essential data, where indices related to semantic content have been designed, generated, and applied so that they are easily manageable[5]-[6]. Once the indices are manually obtained, flexible summarization can be achieved

according to various requests, because indices representing context flow and distinctive events are available. In many cases, though, refinement by a person is costly and time-consuming in many cases, and so automatic indexing remains.

This paper proposes a method that can automatically generate a digest that includes the semantic content of the original video adapting to suit user's preferences, by focusing on context flow and distinctive events. We implemented a system for generating digest from real tennis footage. Although this paper is based on an approach using indices related to semantic content, we introduced domain knowledge and human-action analysis, and limited the kinds of indices, as much as possible, to those that could be obtained by automatic analysis.

The rest of the paper is organized as follows. Section 2 presents the requirements for the summarized video and its indexing process. Section 3 describes the process of identifying important scenes from the indices obtained in section 2 and the process of generating the summarized video and accompanying text adapted to user's preferences. Several experimental results are presented in section 4, and the conclusion is summarized in section 5.

## 2 Requirements for Digest and Acquisition of Semantics

The following four items are considered to be the requirements to generate a digest in this paper. The digest should:

1. express the flow of the whole match,
2. be able to display each memorable, distinctive scene,
3. be able to dynamically reconfigure the content adapting it to the user's preferences, and
4. be generated through internal representation, such as indices, obtained as automatically as possible.

The indices used in this paper are shown as follows:

- score information  $P$ ,
- individual events during play  $A$ .

First, scoring information is extracted from the score region in the video. Analysis methods that can identify the meaning of the telop region in the video and associate it with other media have been previously reported[7]. However, since the test video used was the direct output from the camera without any editing, the scoring information, including the start and end times, were manually prepared in this paper.

Several studies have been reported into indexing methods for detecting individual distinctive moments during play, especially for sports videos[8]-[11]. In this paper, an approach based on domain knowledge and human-action analysis is introduced, since the indices obtained by this approach are expected to be more flexibly associated with the semantic content[10].

Two kinds of indices are automatically extracted as listed in the table 1: play event  $A_t$  to represent the temporal order of each player's actions, and  $A_n$  to indicate distinctive events, such as excellent shots by each player[11].



For example, the event “*serving*” is identified by the following conditions:  
*that both players “stay” at the “backout court” at a certain time, followed by either player doing an “overhead swing” at the “backout court”.*

**Table 1.** Two kinds of play events used

ID	Play event representing temporal order of player’s action	ID	Play event representing player’s best action
0	forehand stroke	0	service ace
1	backhand stroke	1	double fault
2	forehand volley	2	serve & volley
3	backhand volley	3	stroke ace
4	smash	4	smash success
5	serving	5	smash failure
-	—	6	passing success
-	—	7	passing failure

3 Generating Digest

Figure 1 shows a block diagram of the digest generation.

The input data consists of score data  $P$ , video data  $V$ , player’s basic events in time order  $A_t$ , player’s best events  $A_n$ , text elements  $T_e$ , and user’s input  $I$ .

3.1 Generating Structured Video Data

First, structured data  $M$  is generated that hierarchically describes the sets, games, and points for the whole match, the serving player, the point-winning player, basic events of each player, etc. This helps to understand each player’s actions and the status of the match at any given time.

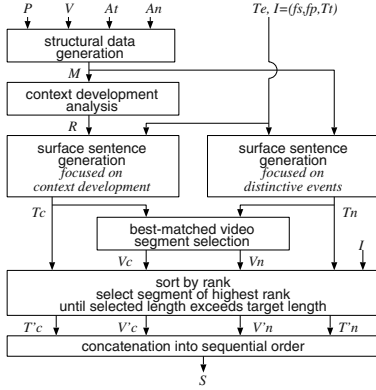
3.2 Acquisition of Context Development

Then, using  $M$ , internal representation  $R$  is generated that describes the overall development of the match. In this paper, the development of the match was analyzed using the value of the player’s superiority and its changes during the match.

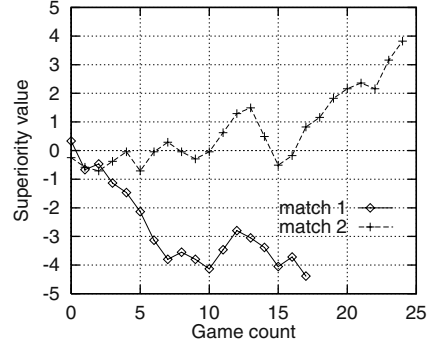
Superiority value  $s$  of a set (or game) indicates how much either player dominated the set (or game), and is calculated as  $s = d_i/d_{max}$ , where

$d_i$  = (actual difference between players in games/points before the last game/point was won in the set/game)

$d_{max}$  = (maximum possible difference between players in games/points before the last game/point was won in the set/game)



**Fig. 1.** Block diagram for generating digest



**Fig. 2.** Overview of match development through superiority value  $S$

For example, the superiority value of a set with the game score 6-2 is calculated as  $s_s = 3/5 = 0.6$ . Likewise, the superiority value of a game won after the second deuce can be obtained as  $s_g = -1/5 = -0.2$ , as the point count is 40-A (in other words, 4-5 in the number of points won). Here, the minus sign shows that the opposite player was dominant during the game. If  $|s_s| \leq s_{s\_th1}$ , the match was recognized as “close”. If  $|s_s| \geq s_{s\_th2}$ , “one-sided”, and if  $s_{s\_th1} < |s_s| < s_{s\_th2}$ , “smooth”, respectively. Currently, the thresholds are set as follows:  $s_{s\_th1} = 0.2$ ,  $s_{s\_th2} = 0.6$ . Likewise,  $s_{g\_th1} = 0.25$ ,  $s_{g\_th2} = 0.6$ .

Let us introduce another superiority value  $S$  obtained by accumulating the superiority value for each game  $s_g$  ( $s_s$  may be used, but  $s_g$  was selected here because  $s_s$  provides a resolution that is too low.):  $S = \sum s_g$ .

As shown in figure 2,  $S$  can be interpreted as an indicator representing the flow of the match until a certain time point. Match 1 shows that the player corresponding to the minus side strengthened his/her hand in the first half, and ended up in a completely one-sided development for him/her in the second half. Likewise, match 2 shows that the balance was maintained in the first half, and, the plus player temporarily got on top until the minus player again regained momentum in the middle, and, the plus player then strengthened his/her hand in the second half.

In summary, internal representation  $R$  describing the development of the match is obtained by determining the superiority value  $s_s, s_g, S$  indicated as a “one-sided”, “close”, or “smooth” development of the game in each set, and by specifying whether these values maintained or changed during the course of the match.

### 3.3 Generating Surface Sentence

Surface sentence  $T$  representing the output narration text is generated using  $R$ , narration text element  $T_e$ , and user input  $I$ .

Here,  $T_e$  denotes the collection of nouns, verbs, adjectives, adverbs, etc. that are sufficient to describe match flow and various player actions. It currently consists of simple tables made up of IDs and various text elements.

For example, if the development flow for the first set was *one-sided*, the following output can be obtained as a surface sentence by referring to  $T_e$ : “*Player Oka won the first set with ease by 6-2.*”

The following example can be obtained as an example related to the best actions by the players: “*He won with great passing shots and a strong serve-and-volley game.*”

In this paper, two kinds of surface sentence are generated:  $T_c$  which relates to the flow of the match represented by  $R$ , and  $T_n$  which relates to the best actions by the players.

In addition, while generating surface sentences, user input  $f_p$  is considered and text elements related to the focus player are prioritized. This focus player is the person or team the user likes and can be selected or ignored in generating the digest depending on preference. Therefore, by using  $f_p$ , the candidate sentences related to the specified player or team can be selected preferentially in the generated narration text.

### 3.4 Acquisition of Video Segments Corresponding to Surface Sentence

Video segments  $V_c, V_n$  typically representing each generated sentence  $T_c, T_n$  are obtained from the relevant range.

In this paper, video segments related to the match flow were chosen from the scenes that included the last hitting event by the player for the last point of a game from each set, and from the scenes that included the most frequent events in every game determined as “*close*” or “*one-sided*”.

Video segments related to important events by players were chosen by ordering the best events of the plays, such as passing shots and service aces, beforehand and by selecting events that were highest in rank, after considering the development flow in the match and the focus player  $f_p$ .

The rank represents a value related to the surface sentence  $T$  and indicates the degree of importance of the text elements within the whole context. Generally, a surface sentence  $T$  has text elements  $T_i$ , and  $T_i$  is of different importance depending on the content. For example, subjects and verbs are higher in rank as these are essential to a sentence. Likewise, modifiers become low in rank.

### 3.5 Determining the Digest

Finally,  $T'_c$  and  $T'_n$  are chosen in descending rank order, to fall in the range of a user-specified duration  $T_t$ . At the same time, corresponding  $V'_c$  and  $V'_n$  are determined. The process stops when the sum of the duration of the selected video segments becomes larger than  $T_t$ . Finally, the digest  $S$  can be obtained after concatenation into sequential order.

Here, another user-input value  $f_s$ , the digest composition, also affects the rankings. The digest composition specifies whether the user wants to see a digest based on match development, or to see a digest focusing on best events such as fantastic shots, or to see a digest that includes both elements. Depending on whether the content of element  $T_i$  is related to match flow or to best events, the weight of the rank attached to element  $T_i$  changes, also varying the temporal portion of each sentence and video segment to be used in the final digest.

## 4 Experimental Results

### 4.1 Results by Proposed Method

Digest video and narration text were generated using several combinations of parameters, such as content composition  $f_s = \{0: \text{focus on match flow}, 1: \text{focus on best events}, 2: \text{both}\}$ , the focus player  $f_p = \{0: \text{player A}, 1: \text{player B}, 2: \text{both}\}$ , and the total time for the digest  $T_t = \{15, 30, 60 \text{ (sec)}\}$ . Narration texts were generated by piecing together the candidate sentences and compensating numerical values, etc. Each candidate sentence had a corresponding video segment.

In the following example, a generated digest is shown with these parameters: content composition = 0, focus player = *Oka*, and total time = 30 (sec).

*"In the first set," "Oka dominated in a one-sided set," "player Oka won the first set with ease by 6-2." "In the second set, though Hinomura temporarily got on top," "it became a close match, and", "player Oka escaped by 6-4."*

Another example of a generated digest is shown below with the parameters: content composition = 1, focus player = *Oka*, and total time = 30 (sec).

*"In the first set, Oka won with a series of passing shots", "and excellent serve-and-volley play, and maintained his lead throughout the set," "player Oka won the first set with ease by 6-2.", "In the second set, Oka won with a service ace", "and stroke ace," "player Oka escaped by 6-4."*

Another example is shown below with the parameters: content composition = 1, focus player = *Hinomura*, and total time = 30 (sec).

*"In the first set, though Hinomura made some excellent shots with great serve-and-volley play", "and several service aces," "player Hinomura lost the first set with ease by 2-6.", "In the second set, although Hinomura made some excellent shots with great passing shots", "and several stroke aces," "player Hinomura was beaten by 4-6."*

### 4.2 Results by People of TV Production Experience

In the proposed method, the three parameters given by a user, i.e. focus player, digest composition, and total duration, are reflected in the content of the generated digest. We asked several people having TV production experience to create correct digest data for reference, which should be generated when the same parameters are given. Table 2 shows the overview of generating digest for reference.

**Table 2.** Overview of generating digest for reference

item	content
creator	5 people with TV production experience including sports
test sequence	3 matches of unedited material video 2 matches of edited broadcast video
parameters	$f_s = \{0,1\}$ , $f_p = \{0,1,2\}$ , $T_t = \{15,30,60\}$

450 digests composed of video and narration were generated using 18 combinations of parameters for each creator and for each test sequence.

In the following example, a generated digest is shown with these parameters: content composition = 0, focus player = *Oka*, and total time = 30 (sec).

*“In today’s friendly match men’s singles, with the opponent Hinomura,”*, *“Oka aggressively took the net from the early stage.”*, *“After he broke the fifth game of the first set with his persistent play, ”*, *“Oka on the bandwagon took the first set on his pace.”*, *“In the second set, though temporarily pushed into a corner by Hinomura,”*, *“Oka steadily piled up the points and won the match straight by 2-0.”*

Another example of a generated digest is shown below with the parameters: content composition = 1, focus player = *Oka*, and total time = 30 (sec).

*“In today’s friendly match men’s singles, with the opponent Hinomura,”*, *“Oka showed fast-moving games with his serve-and-volley play,”*, *“winning all the credit on service,”*, *“and return ace.”*, *“When the opponent took the net, Oka hit through the cross,”*, *“and straight shots.”*, *“Oka maintaining his pace through the match,”*, *“and won straight by 2-0.”*,

Another example is shown below with the parameters: content composition = 1, focus player = *Hinomura*, and total time = 30 (sec).

*“In today’s friendly match men’s singles, with the opponent Oka,”*, *“Hinomura showed fast-moving games,”*, *“scoring points after points by taking the net.”*, *“He put on a sharp performance,”*, *“in serving and retraining,”*, *“making fun of Oka with his outstanding shots.”*, *“However, affected by missed shots at the important places,”*, *“Hinomura lost out the match by 0-2.”*,

### 4.3 Comparison and Discussion

First, let us confirm the qualitative properties of the correct digest generated for reference. Correct narrations had certain structural patterns. Namely, the first sentence introduced the match and the players, the last sentence summarized the result, and the sentence between them described the developments and highlights during the match, demonstrating the logical process of introduction, development, turn and conclusion. This trend was common regardless of test sequences and creators.

Table 3 shows words with a high frequency of use in the first, intermediate, and last sentence of the correct data. It shows that there is a certain correla-

tion between the kinds of words used and the logical structure of introduction, development, turn and conclusion.

Also, it indicates that the intermediate sentences tend to include words representing the match development when  $f_s=0$ , whereas when  $f_s=1$ , they tend to include words representing super plays.

**Table 3.** Words with high frequency of use in correct narration for reference

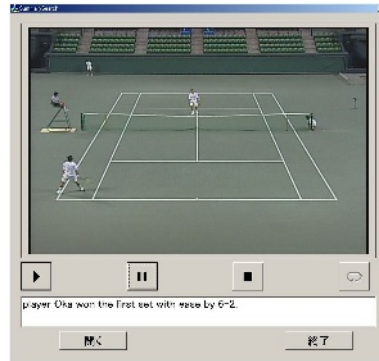
sentence	$f_s=0$	$f_s=1$
first	match, singles, friendly, open, women, men, Japan, mix, doubles	match, singles, friendly, women, versus, open, men, Japan, mix
intermediate	set, game, win, service, however, form, point, lead, drop, miss, play, take, shot, become, in, finally, net	set, service, shot, however, take, game, play, ace, opponent, straight, volley, service, stroke, win, score, net-play, -hand
last	set, do, lose, straight, victory, win, count, finally, brilliant, take	set, victory, lose, match, do, count, straight, take, brilliant, player

The following similarities and differences were confirmed by comparing the digest generated by proposed method with the correct data for reference.

1. The generated digest represented the overall development of the match, but did not contain the introduction and closing remarks as a summary of the match. These elements can be generated by using score data and metadata representing the match and the players.
2. The generated digest favorably reconfigured the content for match development and individual best events, corresponding to the value of  $f_s, f_p, T_t$ .
3. Only the limited expressions were used in the generated narrations. The narrations created the artificial impressions compared to the lively and vivid ones used in the correct data for reference. The generated narrations need to be improved by introducing natural language processing technology, for instance.
4. Some parts of the scenes used in the generated digests were the same as in the correct data for reference, such as the last play of each set, aces, etc., but there were many exceptions. Further comprehensive evaluations through subjective tests would be necessary.

Topics that remain for future work are: fine-tuning the digest generating process, improving the method of evaluating the generated digest, and verifying the effects by using the generated digest for more subjects.

Figure 3 shows an example of playing back the generated digest by the proposed method.



**Fig. 3.** Synchronized playback of generated digest video and narration text

## 5 Conclusion

We proposed a method which can adaptively generate a digest, including the semantic content of the original video, depending on the user's preferences, by focusing on context flow and distinctive events in the video.

Indices for player's basic actions and score information were generated using the player's position, the ball position, and the time point of ball impact from video of actual tennis footage. The system was designed to capture the flow of the whole match and the memorable scenes such as great shots, as the essential components in the generated digest.

The digest generated by proposed method were compared with the correct digest created by the people with TV production experience. Fine-tuning the digest generation process, improving the method of evaluating the generated digest, and verifying the effects by using the generated digest for more subjects remain future work.

## References

1. H. Zhang, et al.: "Automatic Parsing and Indexing of News Video", *Multimedia Systems*, Vol.2, No.6, pp.256-266, 1995.
2. Y. Ariki et al.: "Indexing and Classification of News Video Articles by Speech, Character and Image Recognition", *IEICE, PRMU96-97*, pp.31-38, 1996.

3. M. Smith, T. Kanade: "Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques", IEEE Computer Vision and Pattern Recognition (CVPR), 1997.
4. Y.Nakamura, T.Kanade: "Semantic analysis for video contents extraction - spotting by association in news video", ACM Multimedia, pp.393-401, 1997.
5. K. Ueda, et al.: "A Design of Summary Composition System with Template Scenario", IPSJ, DBS-119-24, pp.139-144, 1999.
6. T. Hashimoto, et al.: "Prototype of Digest Viewing System for Television", IPSJ, Vol.41, No.SIG3(TOD6), pp.71-84, 2000.
7. Y. Watanabe, et al.: "Image Analysis Using Natural Language Information Extracted from Explanation Text", JSAI, Vol.13, No.1, pp.66-74, 1998.
8. Y. Gong, L. T. Sin, C. H. Chuan, H. Zhang, and M. Sakauchi: "Automatic parsing of TV soccer programs", Proc. Int'l Conf. on Multimedia Computing and Systems, pp.167-174, 1995.
9. N. Babaguchi, et al.: "Generation of Personalized Abstract of Sports Video", International Conference on Multimedia and Expo (ICME), FP4.4, 2001.
10. H. Miyamori: "Automatic Annotation of Tennis Action for Content-based Retrieval by Integrated Audio and Visual Information", International Conference on Image and Video Retrieval (CIVR), 2003.
11. H. Miyamori: "Automatic Video Annotation using Action Indices and its Application to Flexible Content-Based Retrieval", IPSJ, DBS127-2, FI67-2, pp.9-16, 2002.



# Content-Based Image Retrieval and Characterization on Specific Web Collections<sup>\*</sup>

R. Baeza-Yates<sup>1</sup>, J. Ruiz-del-Solar<sup>1,2</sup>, R. Verschae<sup>1</sup>, C. Castillo<sup>1</sup>, and C. Hurtado<sup>1</sup>

<sup>1</sup> Center for Web Research, Department of Computer Science, Universidad de Chile, CHILE

<sup>2</sup> Department of Electrical Engineering, Universidad de Chile, CHILE

**Abstract.** One of the challenges in image and video retrieval is the content-based retrieval of images and videos in the web. Less work has been done in this area, mainly due to scalability issues. For this reason, in this paper we investigate this problem by presenting tools for the characterization of the visual contents on specific web collections and a strategy for the search of faces in the web using visual and text information. A case study is also presented in a specific web domain.

## 1 Introduction

Content-based image and video retrieval is a fast growing and increasing relevant research area. The research community recognizes the following main challenges in this field [8]: the bridging of the semantic gap (understanding the meaning behind the query), the content-based retrieval of videos (finding a video similar to another one), and the increasing huge amount of digital data, produced by digital consumer devices (e.g. digital cameras) and computational devices (hard disks, CD-ROMs, etc.), which needs a semantic understanding and also produces a scale problem. In addition to that, we believe that the content-based retrieval of images and videos in the web is an important and challenging area where less research has been done, probably because of technical and practical reasons. As all of us know, popular search engines allow the retrieval of images on the web using only text queries. This situation should be improved and we think we should start to develop methods and strategies for the content-based retrieval of information on the web, the largest and most used multimedia database in the world.

The web is growing at an increasingly rapid pace. More importantly, faster computers and network connections are allowing creators of web content more freedom to add, with fewer constraints, larger quantities of images, graphics, and video. At the same time, people's interest in using images from the web has also increased (the words *pictures* and *pics* are among the most queried terms). Furthermore, given the trend to enrich websites with multimedia, it becomes increasingly important to be able to characterize a given collection of the web according to the multimedia elements that it contains. This type of information is of great importance for Internet service providers (who can determine required levels of

---

<sup>\*</sup> This research was funded by Millenium Nucleus Center for Web Research, Grant P01-029-F, Chile.

regional service), for content producers, and for web search application developers. Characterizing the multimedia contents of the web, however, is a challenging technical problem. First, one must deal with huge amounts of distributed data. Second, it is necessary to use media-specific content-based analysis tools to be able to determine the content of the multimedia elements. With images and video, this means developing tools to automatically determine their visual characteristics: color, texture, shape, etc. More interestingly, it implies using algorithms to automatically detect objects of interest (e.g. persons). Obviously, given the large amounts of data, manual classification is not an option.

In this context, this paper studies the content-based retrieval of images on specific web collections (for practical reasons the whole web can not be studied at the moment), and also the characterization of the visual contents on these collections. For doing that we have developed tools for: efficient web-crawling, content-based image analysis (low-level features such as color, shape and texture), skin segmentation, face detection and web pages' clustering using text information. For developing and testing these tools we have analyzed more than 4 millions web pages, processed more than 383 thousand images (about 35 billion pixels!) and clustered the text of more than two thousand web pages.

This article is structured as follows. Related work is presented in section 2. In section 3 a strategy for the content-based retrieval of faces using visual and text information is proposed, motivating our image characterization results. Tools for processing and analyzing the images of a web collection are described in section 4. In section 5 we present a characterization of the image contents of the .CL domain as a case study. Finally, we conclude in section 6.

## 2 Related Work

The content-based retrieval of images and videos in the web is an underdeveloped area. However, some preliminary work has been done. Two of the most important early works are here outlined. In [9] is presented a system for automatically indexing images collected from the web. Images are automatically collected and assigned to categories based on text surrounding the images. In addition, visual features are extracted from the images to construct a search engine that allows search by visual content. However, the content-based analysis performed in this work is restricted to color histograms. In [4] is implemented a similar system, which in addition uses automatic face detection to index images on the web. This work differs from ours in the specific processing tools being used (our skin and face detection algorithms are much faster), and also in the fact that for solving actual technical problems (bandwidth, response time, etc.), we split the retrieval process in two: the off-line creation of the image database for a specific web collection and the on-line retrieval of the images. Concerning web characterization, to our knowledge, there have not been any studies of web content that use content-based features to characterize the images on the web. In the work of [9] for example, over 500,000 images and videos were catalogued, but general statistics on the visual content of the images in the entire collection (or a subset of the collection using a pre-defined criteria such as our .CL domain) were not presented. Finally, the first version of our characterization study

was presented in a regional conference [5]. In that study only 83,000 images were employed, text information was not analyzed and no distinction was done between home page images and inner page images. All this processing is performed in the current version of this study, citing the mentioned work for the low level feature image analysis.

### 3 Towards a Face Search Engine

As established in the introduction, the content-based retrieval of images and videos in the web is an important and challenging task that should be addressed. However, due to technical limitations (bandwidth, storage capacity, processing time, etc.) a general retrieval system of images for the web cannot be build at this time. This doesn't means that nothing can be done for the moment. On the contrary, this task can be addressed in an incremental way. To start we propose: (i) to restrict the domain of operation of the retrieval system to a certain web collection to build a vertical image search engine. We have chosen to work on the .CL domain, whose characteristics and dimensions allow the implementation of a prototype; (ii) to create an image database where the search process will be carried out. This database (a cache of images) is created off-line, using the crawling tool described in section 4.2, for solving the problems related with the required time for the gathering of the images. Thus, the on-line process of image retrieval is performed on this database; (iii) to filter the images to be stored in the database according with the functionality of the retrieval system to be built (for dealing with the storage capacity limitations). In our case we want to build a person search engine; this means that graphics, images non-containing skin and images non-containing faces should be filtered, in addition with repeated images, all these filters are described in section 4; (iv) to process and label the images to be stored in the database. In the case of the person search engine that means to store the position of the faces detected in each image and the web page class of the text associated with this image (the page clustering algorithm employed is described in 4.6).

Webfaces, the person search engine under construction, is based on the use of face and text information. For searching a given person, the user should provide a picture of the person and optionally a related text (a group of keywords). The search system will provide a set of database images where the person can be present, and a confidence value for each image. The text information will be used to determine the associated web page class and therefore to restrict the search process to a given portion of the database (the set of images with this associated class). The face contained in the provided picture is used to do a similarity search with all the faces containing in the selected subset of the database using a face recognition algorithm. Using this information, the text clustering information will be used to recommend clusters related to the similar images as well as keywords that can help to improve the query. All relevant subsystems of Webfaces are already built. We are working on the implementation of fast similarity algorithms based in metric spaces and on solving scale and orientation issues of the face recognition process, to finish the integration of all the mentioned components.

## 4 Tools for Analyzing the Images of a Web Collection

### 4.1 Proposed Methodology

For developing and testing the image retrieval and analysis tools, which are the same tools employed for characterizing the contents of certain web collections, we employed real web data (images and text) sampled from the .CL top level domain (4 millions web pages, 383,000 images and the text of 200,000 web pages containing the selected images.). The processes employed for obtaining and processing this data are: (i) web-crawling for sampling a given web collection and obtaining image links (I-URLs) and web pages associated with these images, i.e. the web pages where the images are found (W-URLs); (ii) color, edge and texture low-level visual analysis for characterizing different kinds of images and for constructing image filters, such as photographs vs. graphics and indoor vs. outdoor; (iii) skin segmentation algorithms for detecting image areas where humans and human-body parts are present; (iv) face detection algorithms for detecting humans; and (v) tools for clustering web pages using the text information associated with the processed and selected images.

### 4.2 Web-Crawling

Our web-crawling architecture is based on a long-term schedule for collecting sites and a short-term schedule that worries about network politeness and use of resources (CPU, bandwidth) [1]. First we obtain a list of the domains of interest (all the domains registered under .CL) and then we use our crawler to obtain the web pages in each of the selected domains. The next step consists of automatically extracting the links to the images (I-URLs) and the links to the associated web pages (W-URLs). For practical purposes (processing time and storage capacity) the total amount of links is sampled and a statistical representative subset of them is employed for the developing and testing of the tools. The crawling of the .CL domain was performed in May 2003, August 2003 and January 2004. Each time about 1.3 million web pages were analyzed and the downloaded images were 100,000 in May 2003, 83,000 in August 2003 and 200,000 in January 2004. Text information was processed only in January 2004, and the total amount of web pages downloaded for this processing was 200,000.

### 4.3 Low-Level Visual Analysis

A set of 72 visual features that represent color, shape and texture was extracted (see feature description in [5]). Although some of these features are fairly simple, they are useful in giving a snapshot of the visual content of images in the web and in the construction of image filters. Using these basic features we build a photograph vs. graphics filter. This filter was implemented using a support vector machine classifier and 5 from the 72 features (aspect ratio, standard deviation in the R histogram, average of the S component, percentile 90% in the R histogram and the texture feature LD in 0°), which were automatically determined using forward selection [13]. The performance of the obtained classifier is 94.5%.

#### 4.4 Skin Segmentation

This functionality was implemented using *SkinDiff* [7], a robust skin segmentation algorithm that uses neighborhood information. The decision about the pixel's class is taken using a spatial diffusion process that employs context information. In this process a given pixel will belong to the skin class if and only if its Euclidean distance, calculated in a given color space, with a direct diffusion-neighbor that already belongs to the skin class, is smaller than a certain threshold ( $T_{diff}$ ). The seeds of the diffusion process are pixels with a high probability of being skin, i.e. the skin probability is larger than a certain threshold ( $T_{seed}$ ). The extension of the diffusion process is controlled using a third threshold ( $T_{min}$ ), which defines the minimal probability allowed for a skin pixel. *SkinDiff* uses the RGB color space (normally images in the web use this color space) and a *Mixture of Gaussians* (MoG) model for determining the skin probabilities. For a fast computation, the MoG is implemented using look up tables (LUTs). It is not necessary to store the skin probabilities in the LUT, but only the information concerning the following three situations: skin probability larger than  $T_{seed}$ , smaller than  $T_{min}$  or in  $[T_{seed}, T_{min}]$ . Therefore for each possible RGB combination, only 2 bits needs to be stored. For an adequate implementation of the LUTs, the colors in each channel are quantized to 64. Using *SkinDiff* a 320x280 image is processed in about 0.2 seconds.

#### 4.5 Face Detection

This algorithm detects frontal faces with small in-plane rotations. The detector corresponds to a cascade of filters, where each filter discard non-faces and let face candidates pass to the next stage of the cascade. This architecture seeks to have a fast detection, considering the fact that only a few faces are to be found in an image, while almost all of the image area corresponds to non-faces. This fast detection is achieved in two ways: (i) having a small complexity in the first stages of the cascade, and (ii) using simple rectangular features (the filters), which are quickly evaluated using a representation of the image called the integral image [12]. Each of the filters of the cascade is trained using the Adaboost classifier [12]. The images are analyzed using 24x24 pixel windows. Each window corresponding to a color image is preprocessed (filtered) using the skin segmentation algorithm described in 4.4. The number of skin pixels in each window is counted, and if this number is smaller than 50% of the pixels of the window, then this window discarded, otherwise, it is further processed. With this procedure, face detection time was reduced by a factor of 2 and the number of false detections was reduced considerably with an increase in the face detection rate. The increase in the detection rate was achieved by reducing the number of stages in the cascade when the detector was applied to color images (in gray scale images 49 stages were used, while in color images only 42). Additionally, the cascade processing was complemented using a statistical classifier added in parallel at the end of the cascade. The idea behind this procedure is the following: when fewer stages in the cascade are implemented, the detection rate increases but the false detection rate also rises (remember that each cascade stage filters non-face windows). On the other hand, a statistical classifier of face and non-face windows, implemented using color and texture low-level features, decreases the detection rate of the cascade, but also the

false detection rate. Thereafter, a best compromise can be found between the obtained detection rate and false detection rate, by placing the statistical classifier at its end. After many trials it was found that the best place to put the classifier was after the stage number 35. The selected classifier was the SVM and the low-level features determined using forward selection [13] were average of the B channel, standard deviation of the G channel, average V component, number of colors greater than 2% of image area, percentile 50 of the G channel, percentile 10 of the B channel, and the number of edge pixels in 45° greater than the average edge of the window. Finally, the obtained detections (detected face regions) are fused for determining the size and position of the final detected faces. Overlapping detections are processed for filtering false detections and for merging correct ones. All detections are separated in disjoint sets using the heuristic described in [11].

#### 4.6 Text Clustering

Images in the web are inserted into web pages using the IMG html tag. The attribute ALT of the IMG tag allows us to specify a text alternative to the image, which is automatically displayed when the browser cannot display the image. Some images are included within a hypertext anchor: in this case an image may behave as a button linked to other documents or resources. The text in the ALT attribute, along with the text inside the hypertext hidden meanings. This motivated us to use the whole text anchors as candidate descriptors for the image. However, only a small fraction of the images in our collection have such descriptions. Furthermore, the quality of these descriptions is low; many of them have few words which sometime refer to file names with in the web pages as the accompanying text for the images. The text in web pages gives us some approximated context for each image. We left as future work the discovering of better descriptors for images. Such task may consider heuristics for extracting data from anchor text, ALT tags, or other parts of the html page that includes the image. We ran a clustering process over text in web pages of the images. Our goal is to discover clusters that define textual contexts for the images. Such clusters are the basis in our approach for integrating textual contexts in our image retrieval tool. Clusters centroids can be used to model textual contents, and user queries, specified as list of terms, can be compared against the centroids to determine the relevant contexts users are searching for. When the cluster associated to a query is found, the search for images can be focused on the images of the cluster. The clustering process is achieved by an implementation of a k-means algorithm provided by the clustering toolkit CLUTO [3]. We used a k-means algorithm for its simplicity and low computational cost. In addition, it has proved to be very effective for clustering collection of documents [14].

### 5 Case Study: Characterization of the Images of .CL Domain

Due to the limited extension of this article in this section we present a very condensed part of our characterization results. The complete results, including histograms, graphics and a complete statistical analysis can be found in our website [15].

## 5.1 Crawling

**Domains and pages.** In the most recent official study of the .CL domain [2] almost 2 million pages were found in 38,307 sites in 34,867 domains. Current estimations of TodoCL [10] point out that the Chilean Web has 5 millions of pages  $\pm 10\%$  and that the number of sites and domains is  $80,000 \pm 10\%$ . From the 3.5 million pages used for this final version of our study (we considered up to 4 levels of links), we obtained two samples: one of home pages and one of inner pages.

**Home Page Images.** This collection was obtained from 36,455 home pages; from those home pages, 23,523 had objects or links to non-textual URLs. In total 338,963 links were found, 208,066 of them unique. From the unique links, 60.0% were to GIF images, 26.8% to JPG images, 7.7% to Flash animations, 2.6% to style-sheets, and 0.7% to PNG images; the rest was mostly to PDF or Word documents. The total number of GIF, JPG and PNG images was 183,669, from those, 100,000 were randomly selected.

**Inner Page Images.** The sample of inner pages was obtained in 8 hours of crawling, with 443,000 pages downloaded. We discarded all the pages that were at depth greater or equal to 5 in the websites, and all the pages without links to images, obtaining a sample of 311,589 pages. We believe that this sample is representative of what a user sees while browsing the web; and using pages at deeper levels would bias the sample towards large, dynamic websites. These pages contained 9,148,115 links to images, and only 926,781 were unique, relatively much less unique links than in the home page collection. Our interpretation is that web site owners usually have a small set of images, which are repeated across their entire websites. From the unique links, 53.9% were to GIF images, 35.4% to JPG images, 2.8% to Flash animations, 2.2% to style-sheets, and 0.8% to PNG images. There is a significant diminution of animations in the inner pages. The total number of GIF, JPG, and PNG images was 842,902. From those, 100,000 were randomly selected.

## 5.2 Image Processing and Analysis

**Visual Features.** We extracted all 72 visual features mentioned in section 4.3 from the 200,000 images that were processed. It was found that 19.2% of the images correspond to photographs and the rest to graphics. It is interesting to mention that the number of images with a certain area follows a Power law distribution. The analysis was split between home pages images (HP-images) and inner pages images (IP-images).

**Skin Detection.** It was detected skin in the 6.5% of the HP-images and in the 7.9% of the IP-images. The reason of these different percentages seems to be the larger size of IP-images and the larger proportion of photographs in this set. The average size of the skin clusters is 3167/3121 pixels, and the mean number of skin cluster in each image is 3.73/4.14, for the HP- and IP-images, respectively.

**Faces Detection.** We found that 2.07% of the HP-images, while 2.12% of the IP-images contained faces. The average number of faces per image (from those images containing faces) is 2.1167/2.1162 for the HP- and IP-images, respectively. The

maximum number of faces found in a single image was 89/39 for the HP-/IP-images<sup>1</sup>. It was also found that the distribution of the number of faces in both image sets (considering only the images that contain faces) is close to a Power law. For example, for HP-images considering cases from 2 to 10 faces, the parameter of the distribution is -2.13.

### 5.3 Text Analysis

We consider two sets of 1,965 images each, corresponding to images that arise in web pages. The first set has images with high probability of having portions of human skin, and the second set contains images with human faces. The *face* images belong to 1,480 web pages, while the *skin* images belong only to 748 different web pages. These images were obtained using the algorithms already described. We model the associated web pages as term-weight vectors using a vocabulary of 20,600 words. *Stopwords* were eliminated from this vocabulary. The cosine similarity function between vectors was employed.

The clustering process is guided by a score function that measures the overall clusters quality. The score function used is the total sum of the average similarities between the vectors and the centroids of the clusters that are assigned to. Each run of the algorithm computes  $k$  clusters. Thus, in order to study adequate values of  $k$  we run the algorithm several times. We reach a quality score of 0.80, which reflects high similarities between objects in each clusters, at  $k=250$  and  $k=300$  for the home-skin and home-face dataset, respectively. In [15], we show some figures that depict the quality of the clusters found for different values of  $k$ , for the *face* and *skin* web page sets. These figures also show curves with the incremental gain of the overall quality of the clusters, and a histogram for the number of clusters per average intra-cluster similarity for the two datasets at the aforementioned values of  $k$ . Many clusters that represent clearly defined contexts for images were found. In [15], we present also tables with some of the found clusters. These clusters allowed us to discover semantic connections between web pages having faces. An example of that is a cluster containing web pages related to movies, musicals, DVDs, etc. Also, good search keywords can be detected using word frequency for each cluster.

### 5.4 Processing Time

*Page Gathering*: It took about 8 hours for the 400,000 pages using a single, standard PC running Linux. With this setting, the page recollection of the whole Chilean web takes about five days. *Feature Extraction*: The process of automatic extraction of the 72 visual features on the 200,000 images under analysis takes about 47 hours on a single, standard PC running Linux. *Skin Segmentation* and *Face Detection*: The process of skin segmentation and face detection on the 200,000 images took about 10 hours and 40, respectively, using a single, standard PC running Linux. *Webpage Clustering*: The text of 2228 web pages was clustered. It took 5 minutes to compute 300 clusters using a standard PC, running Windows XP. Obviously, any of these

---

<sup>1</sup> A group photo in [www.bradford.cl](http://www.bradford.cl) has 89 faces!



processes can be speeded up using more than 1 PC, and can be done in a reasonable time, as is an off-line task.

## 6 Conclusions

We investigated the content-based retrieval of images on specific web collections and also the characterization of the visual contents on these collections. For doing that we presented tools for: efficient web-crawling, content-based image analysis (low-level features such as color, shape and texture), skin segmentation, face detection and web pages' clustering using text information. For developing and testing these tools we analyzed more than 4 millions web pages, processed more than 383,000 images and clustered the text of more than 2,200 web pages. A first application of these tools is the characterization of the image contents of the .CL domain. For carrying out this study a statistical representative subset of the total number of images of the .CL domain was employed. In the final version of this article we plan to also include results on clustering a larger sample of only the text segments surrounding the selected images.

In this article we also presented a strategy for the content-based search of persons using visual and text information is proposed. All relevant components of this system, including a face recognition subsystem, are already built. We are working in the system final integration, which will be reported in a near future.

## References

- [1] R. Baeza-Yates, and C. Castillo, Balancing collection volume, quality and freshness in a web crawler, in A. Abraham. J. Ruiz-del-Solar, M. Köppen (Eds.), *Soft-Computing Systems: Design, Management and Applications, Frontiers in Artificial Intelligence and Applications* 87, IOS Press, pp. 565 – 572, 2002.
- [2] R. Baeza-Yates, B.J. Poblete, and F. Saint-Jean, *Evolución de la Web Chilena 2001-2002 (Evolution of the Chilean Web 2001 - 2002)*, Center for Web Research, Department of Computer Science, Universidad de Chile, January 2003 (in Spanish).
- [3] CLUTO Home page: <http://www-users.cs.umn.edu/~karypis/cluto/>
- [4] C. Frankel, M.J. Swain and V. Athitsos, *WebSeer: An Image Search Engine for the World Wide Web*, University of Chicago Technical Report TR-96-14, July 31, 1996.
- [5] A. Jaimes, J. Ruiz-del-Solar, R. Verschae, D. Yaksic, R. Baeza-Yates, E. Davis, and C. Castillo, On the Image Content of the Web in Chile, *Proc. of the First Latin American Web Congress*, IEEE CS Press, 72 – 83, Santiago, Chile, Nov. 10 – 12, 2003.
- [6] Y. Rui, T.S. Huang, and S.-F. Chang, Image Retrieval: Current Directions, Promising Techniques, and Open Issues, *Journal of Visual Communication and Image Representation*, No. 10:1-23, 1999.
- [7] J. Ruiz-del-Solar and R. Verschae, Robust Skin Segmentation using Neighborhood Information, ICIP 2004, submitted.
- [8] N. Sebe, M. Lew, X. Zhou, T. Huang and E. Bakker, The State of the Art in Image and Video Retrieval, *Lecture Notes in Computer Science* 2728 (*Image and Video Retrieval 2003*) 1 – 8, 2003.

- [9] J.R. Smith and S.-F. Chang, An Image and Video Search Engine for the World-Wide Web, *Proc. of SPIE Storage & Retrieval for Image and Video Databases V*, Vol. 3022, pp. 84-95, San Jose, CA, Feb. 1997.
- [10] TodoCL Search Engine (<http://www.todocl.cl/>)
- [11] R. Verschae and J. Ruiz-del-Solar, A Hybrid Face Detector based on an Asymmetrical Adaboost Cascade Detector and a Wavelet-Bayesian-Detector, *Lecture Notes in Computer Science* 2686, Springer, 742-749, 2003.
- [12] P. Viola and M. Jones, Fast and Robust Classification using Asymmetric AdaBoost and a Detector Cascade, *Advances in Neural Information Processing System* 14, MIT Press, Cambridge, MA, 2002.
- [13] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann, 1999. Weka homepage: <http://www.cs.waikato.ac.nz/~ml/weka/>
- [14] Y. Zhao and G. Karypis, Comparison of Agglomerative and partitional document clustering algorithms, *SIAM Workshop on Clustering High-dimensional Data and its Applications*, 2002.
- [15] <http://www.cwr.cl/chile-images/>

# Exploiting Problem Domain Knowledge for Accurate Building Image Classification

Andres Dorado<sup>1,2,\*</sup> and Ebroul Izquierdo<sup>2</sup>

<sup>1</sup> Pontificia Universidad Javeriana, School of Engineering,  
Calle 18 #118-250 Cali, Colombia

<sup>2</sup> Queen Mary, University of London, Electronic Engineering Department,  
Mile End Road, London E1 4NS, UK  
{andres.dorado, ebroul.izquierdo}@elec.qmul.ac.uk

**Abstract.** An approach for classification of building images through rule-based fuzzy inference is presented. It exploits rough matching and problem domain knowledge to improve precision results. This approach uses knowledge representation based on a fuzzy reasoning model for establishing a bridge between visual primitives and their interpretations. Knowledge representation goes from low level to high level features. The knowledge is acquired from both visual content and users. These users provide the interpretations of low level features as well as their knowledge and experience to improve the rule base.

Experiments are tailored to building image classification. This approach can be extended to other semantic categories, i.e. skyline, vegetation, landscapes. Results show that proposed method is promising support for semantic annotation of image/video content.

## 1 Introduction

The evolution of multimedia applications from the recent past to the near future goes from automatic signal extraction to user-provided knowledge, passing throughout features and semantics. A critical paradigm that has captured attention of researchers is “bridging the gap”, i.e. from features to semantics[1]. This problem can be stated as “to find a technique for automatic recognition of the underlying semantic structure of given features”.

In general terms, annotation is a process to represent features by symbols. In the context of semantic annotation these symbols represent user interpretations of the features. The process of assigning a feature vector to a specific symbol is a classification task. Thus, the annotation process is decomposed into feature extraction followed by classification.

In this work, an approach for semantic classification of building images through rule-based fuzzy inference is presented. It uses knowledge representation based on a fuzzy reasoning model for establishing a bridge between visual primitives and their interpretations.

---

\* The research leading to this paper was done within the framework of the Network of Excellence SCHEMA.

This approach exploits rough matching, problem domain knowledge and the effect of weighted rules in fuzzy rule-based classification to improve precision results. Consequently, a high classification performance can be achieved with a relatively simple and consistent model which goes in the way human reasoning works. [2] presents a detailed analysis of rule weight effects in classification systems.

The approach is tailored to annotate building images. Existing approaches for classification of building images use a bayesian framework to exploit image features by perceptual grouping [3], binary bayesian hierarchical classifiers [4], or perform building semantic extraction using support vector machines [5].

Following, Section 2 defines the classification problem. Afterwards, Section 3 presents the approach. In Section 4 knowledge representation for building image classification is described. Section 5 gives a summary of experimental results. Finally, Section 6 concludes the paper.

## 2 Problem Definition

This work addresses the problem of classification of low level features into high level concepts as a first step towards semantic image/video annotation.

Let  $\mathbf{x} = (x_{11}, x_{12}, \dots, x_{1N}, x_{2N}, \dots, x_{MN})$  be an image,  $\mathbf{f} = \{\mathbf{f}^{(1)}, \dots, \mathbf{f}^{(V)}\}$  be feature sets where  $\mathbf{f}$  is a function of the image  $\mathbf{x}$ ,  $E^v = (e_1^v, \dots, e_L^v)$  be a pattern extracted from a feature vector  $\mathbf{f}^{(v)}$ , and  $Y = \{y_1, \dots, y_K\}$  be a class set. The classification problem is stated as: Learn a function

$$g(\mathbf{x}) : E^v \mapsto Y, \quad (1)$$

where  $y_j$  are symbols identifying categories and representing semantic interpretations of pattern  $E^v$ .

$g(\mathbf{x})$  can be decomposed into  $K$  single-class specialised classifiers

$$g_j(\mathbf{x}) : E^v \mapsto y_j, \quad 1 \leq j \leq K. \quad (2)$$

Subsequently, a fuzzy model is extracted from feature set  $\mathbf{f}^{(v)}$  in order to approximate each function  $g_j(\mathbf{x})$  by a set  $R_j = \{R_{j1}, \dots, R_{jC}\}$  of  $C$  *if-then* rules.

Therefore,  $g(\mathbf{x})$  is summarised by a rule base  $R$  as follows:

$$g(\mathbf{x}) \approx R = \bigvee_{j=1, k=1}^{K, C} {}^w R_{jk}, \quad (3)$$

where  $w \in [0, 1]$  is the weight of rule  $R_{jk}$ .

## 3 A Semantic Annotation Process

Based on [6] the proposed image annotation process denoted by  $L$  can be summarised as

$$\{V^0, \mathbf{x}_i, Y\} L \{V^n, z_i\}, \quad (4)$$

where  $V^0$  is an abstract non annotated image space defined as a tuple  $\{X, Y\}$ ,  $X$  is a set of images,  $Y$  is a set of symbols, and  $\mathbf{x}_i \in X$  is a selected image from a video unit to be annotated.

The process  $L$  begins by creating an instance of the abstract non annotated image space  $V^0 = \emptyset$ . Afterwards, the annotation operator  $l \in L$  maps  $V^0$  into the annotated image space in an interactive annotation session, which is a sequence of annotation spaces  $V^0, \dots, V^n$  with  $V^n(\mathbf{x}_i) = l(V^n)$ .

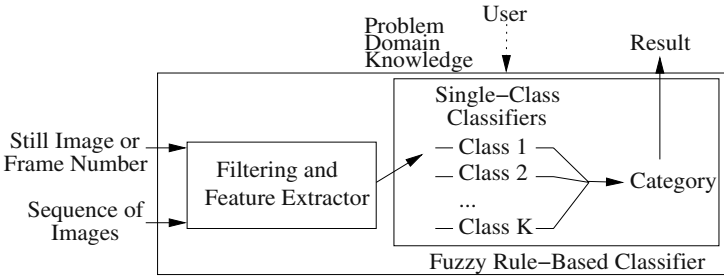
The result is a set of labels  $z_i = \{z_{i1}, \dots, z_{iM}\}$  which represents annotations spatially and temporally linked to descriptive concepts of the video unit content. Each label has the form

$$z_{ij} : \mathbf{x}_i \mapsto y_j, \quad (5)$$

where  $g(\mathbf{x}_i) = z_{ij}$  and  $y_j$  is a symbol labeling descriptions of image  $\mathbf{x}_i$ .

These descriptions can be either real objects in the image (e.g. building, car, person, etc.) or abstract concepts describing what is happening in the image (e.g. a person is driving a car). On the one hand, classification of real objects is useful for annotation of video units by applying image understanding. On the other hand, classification of abstract concepts is closer to video understanding.

Fig. 1 corresponds to an overview of the process. Firstly, a pre-processing step performs filtering and selection of suitable low level features from a training set  $X$  in order to facilitate pattern extraction. [7] presents a pre-processing procedure on sequence of images extracted from videos for improving building image classification.



**Fig. 1.** Data flow of the approach for semantic annotation

Afterwards, a rule-based fuzzy model is extracted from feature set  $\mathbf{f}^{(v)}$  to associate low level features with each specific class represented by a particular symbol  $y_j$ . This step corresponds to the training of single-class classifiers. In addition to the information extracted from sample images, the fuzzy model is improved with problem domain knowledge provided by a user.

The result is a classification model based on a fuzzy reasoning and used for establishing a bridge between visual primitives and their interpretations [8]. Finally, classification results are used by process  $L$  as annotations of image  $\mathbf{x}_i$  with symbol  $y_j$ .

## 4 Building Image Classification

This approach is tested using a standard visual descriptor whose main characteristic is the simplicity to represent local and global distribution of edges without expensive computation. It uses luminance mean values to detect edges that is considered a weak approach. However, weakness of low-level description is compensated with contributions of problem domain knowledge given by a user. Contributions can be performed on-line. Thus, fine-tuning is possible through relevance feedback.

Section 4.1 presents the semantics of the descriptor used to extract salient features from building images and Section 4.2 gives details of classification model.

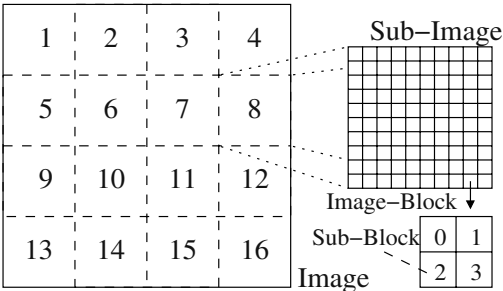
### 4.1 Feature Space

Feature space is built with feature vectors extracted from image database using the MPEG-7 *edge histogram descriptor*[9]. This visual descriptor uses 16 histograms to represent local distribution of directional edges within an image. These histograms consist of bins associated with five edge categories namely horizontal, vertical,  $45^\circ$ ,  $135^\circ$ , and nondirectional edge.

The image is spatially decomposed into 16 sub-images using a fixed grid with equal-size rectangles. Each sub-image is divided into a given number of non-overlapping small square blocks (image-blocks). Consequently, the block size depends on the sub-image size. The basic components of edge histogram descriptor are shown in Fig. 2.

Blocks are also divided into 4 sub-blocks. The luminance mean values in the gray scale is measured in order to determine the sub-block pixel intensity. Then, blocks are passed through five masks to assign the corresponding edge category.

The number of blocks per edge category is counted to compute the edge distribution within a sub-image. The bins in the histograms summarise the distribution of each edge category of the 16 sub-images in a left-right and top-down scanning.



**Fig. 2.** Basic components of edge histogram descriptor

The semantics of edge histogram descriptor can be summarised as

$$ehd(\mathbf{x}) = \{h_1^1, h_1^2, \dots, h_1^5, h_2^1, \dots, h_i^j, \dots\}, \quad (6)$$

where  $\mathbf{x}$  is an image,  $h_i^j$  is the number of edges of category  $j$  in the  $i^{th}$  sub-image.

Therefore, feature space consists of feature vectors with 80 dimensions.

## 4.2 Classification Model

According to semantics of *edge histogram descriptor* (Eq. 6) five input variables are defined for the fuzzy model. Each input variable corresponds to an edge type category.

A compact fuzzy model is proposed associating a group of fuzzy sets with each input variable. In order to simplify the model three fuzzy sets are used by default. Linguistic hedges are not applied on this model. Three standard piecewise linear functions are used to determine memberships to these fuzzy sets: Z-shape,  $\Lambda$ -shape, and S-shape. These functions are adjusted to specific intervals on x axis called *boundaries*. Intervals are computed using a clustering technique, i.e. fuzzy c-means. In Tab. 1 presents a summary of the fuzzy variables with their fuzzy sets and boundaries.

**Table 1.** Fuzzy variables used for Edge Histogram Descriptor. Each fuzzy set, e.g. Low, Medium, or High, is defined by a membership function. These functions have a domain between the interval determined by the corresponding boundaries  $b_1$ ,  $b_2$ , and  $b_3$ . Intervals are automatically generated using Fuzzy C-Means algorithm

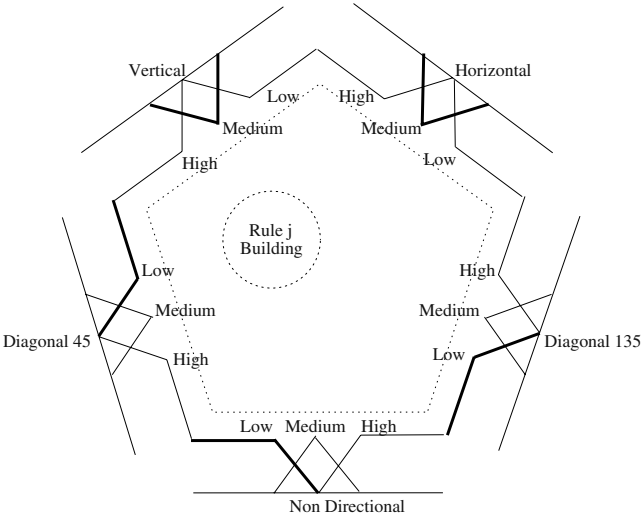
Universe (Variables)	Fuzzy Set	$b_1$	$b_2$	$b_3$
Vertical Edge,				
Horizontal Edge,	Low	0.1	0.3	1.0
Diagonal 45°,	Medium	0.1	0.3	0.4
Diagonal 135°,	High	0.0	0.3	0.4
Non Directional				

Classification model combines input variables with labels through fuzzy inference rules which are used to classify new images into the defined semantic categories. These inference rules are organised into a multi-input single-output rule space as is shown in Fig. 3.

Each inference rule has an IF-part with five antecedents and a THEN-part with one consequent as follows:

$$\text{IF } e_1^v \text{ is } A_1^k \dots \text{AND } e_5^v \text{ is } A_5^k \text{ THEN classify as } y_j, \quad (7)$$

where  $e_i^v$  is an instance of an input variable (edge type),  $A_i^k$  is a linguistic term (fuzzy set name) used to transform values from a continuous to a discrete domain, and  $y_j$  is a symbol labeling a semantic category. Classifier uses rule base for labeling features as either “Non Building” or “Building”.



**Fig. 3.** Rule base. Fuzzy variables associated with each type of edge are combined in a multi-input single-output space.  $R_j$  is an *if-then* rule as is shown in Eq. 7. Fuzzy sets included in rule  $R_j$  are indicated with highlighted functions

## 5 Experimental Results

The classification approach is tailored to the semantic category “Building”. Subjective selection of characteristics required to classify an image into this category is performed.

An image is categorised as “Building” when a building structure of a visible size appears in the scene. One shortcoming in this type of images is the different kinds of buildings, i.e. castles, houses, warehouses, religious facilities, etc. In Fig. 4 some randomly selected samples of building images is illustrated.



**Fig. 4.** Samples of building images

Experiments were conducted using a testing set with over 3000 distinct images extracted from TRECVID[10] video repository. For training 15 and 100 building images were selected from TRECVID and Corel Draw Gallery, respec-



tively. A set of images (key-frames) extracted from videos was automatically classified using the proposed approach.

The ground truth for all images was assigned by a single subject. Classifier performance evaluation is based on the amount of images correctly classified and the number of misclassified images. The classification results are given in the Tab. 2.

A first run (**test 1**) was performed to evaluate classification results after applying local analysis. As *edge histogram descriptor* decomposes original images into 16 sub-images, local analysis means classification of each single sub-image. Salient edges found in these sub-images and satisfying criteria of inference rules are classified as “Building”. This kind of analysis allows detection of parts of a building structure within the scene. Classification result is over 70%.

A second run (**test 2**) was performed to evaluate classification results after applying global analysis. It means that an image is classified as “Building” when the number of sub-images classified as “Building” is greater than a threshold. This kind of analysis allows reducing of misclassification introduced by sub-images satisfying conditions but not being part of a building structure.

A third run (**test 3**) was used to evaluate classification results after varying the rule weights. These weights are real values in the range  $[0, 1]$  as is indicated in Eq. 3. It means different weights are assigned to each rule considering its relevance to determine the pattern matching of a sub-image. These weights are modified by the user (problem domain knowledge). Experiments show this procedure improves classification results.

**Table 2.** Classification results [ % ]

Test	Correctly Classified	Misclassified
1	70.05	29.27
2	83.95	14.51
3	86.31	13.29

## 6 Conclusions and Further Work

An approach for building image classification that utilises a fuzzy reasoning model is presented. It decomposes the process in feature extraction and classification. Single-class classifiers are used to assign images into semantic categories defined by the user.

This approach exploits rough matching and problem domain knowledge to get high classification performance even with few sample images. Combining few training images with contributions provided by a user 86% of precision was obtained classifying over 3000 images extracted from real-world videos.

Experiments are tailored to building image classification. However, this approach can be extended to other semantic categories, i.e. skyline, vegetation, landscapes, etc.

This approach was tested using a standard visual descriptor which main characteristic is the simplicity to represent local and global distribution of edges without expensive computation. Weakness of low-level descriptions is compensated with contributions of problem domain knowledge give by a user.

The fuzzy reasoning model facilitates identification of participating rules in a specific classification result. Rule weights can be used either to enable or to disable specific rules. This tuning actions can be performed on-line.

Further work is addressed to extract the classification model from image database using unsupervised learning. Afterwards, the model is tuned using relevance feedback.

## References

1. Dorai, C. and Venkatesh, S.: Bridging the Semantic Gap with Computational Media Aesthetics. *IEEE Multimedia*, Vol. 10. **2** (2003) 15–17.
2. Ishibuchi, J. and Nakashima, T.: Effect of Rule Weights in Fuzzy Rule-Based Classification Systems. *IEEE Trans. on Fuzzy Systems*, Vol. 9. **4** (2001) 506–515.
3. Iqbal, Q. and Aggarwal, J. K.: Applying Perceptual Grouping to Content-Based Image Retrieval: Building Images. In: *Proc. IEEE Int'l Conf. on Computer Vision and Pattern Recognition (CVPR99)*, Vol. 1. (1999) 42–48.
4. Vailaya, A., Figueiredo, M. A. T., Jain, A. K. and Zhang, H-J.: Image Classification for Content-Based Indexing. *IEEE Trans. on Image Processing*, Vol. 10. **1** (2001) 117–130.
5. Wang, Y-N., Chen, L-B. and Hu B-G.: Semantic Extraction of the Building Images using Support Vector Machines. In: *Proc. IEEE Int'l Conf. on Machine Learning and Cybernetics*, Vol. 3. (2002) 1608–1613.
6. Dorado, A. and Izquierdo, E.: Semantic Labeling of Images Combining Color, Texture and Keywords. In: *Proc. IEEE Int'l Conf. on Image Processing (ICIP 2003)*, Vol. 3. (2003) 9–12.
7. Zeljkovic, V., Dorado, A. and Izquierdo, E.: A Modified Shading Model Method for Building Detection. *5th Int'l Workshop on Image Analysis for Multimedia Interactive Services, (WIAMIS 2004)*, to appear. (2004).
8. Dorado, A., Calic, J. and Izquierdo, E.: A Rule-Based Video Annotation System. *IEEE Trans. on Circuits and Systems for Video Technology*, to appear. (2004).
9. Choi, Y., Won, C. S., Ro, Y. M. and Manjunath, B. S.: Texture Descriptors. In: *Manjunath, B. S., Salembier, P. and Sikora, T. (eds.): Introduction to MPEG-7, Multimedia Content Description Interface*. Chapter 14. Wiley (2002) 213–229.
10. TRECVID: TREC Video Retrieval Evaluation.  
<http://www-nlpir.nist.gov/projects/trecvid/>. 2003

# Natural Scene Retrieval Based on a Semantic Modeling Step

Julia Vogel and Bernt Schiele

Perceptual Computing and Computer Vision Group

ETH Zurich, Switzerland

{[vogel](mailto:vogel@inf.ethz.ch),[schiele](mailto:schiele@inf.ethz.ch)}@inf.ethz.ch

<http://www.vision.ethz.ch/pccv>

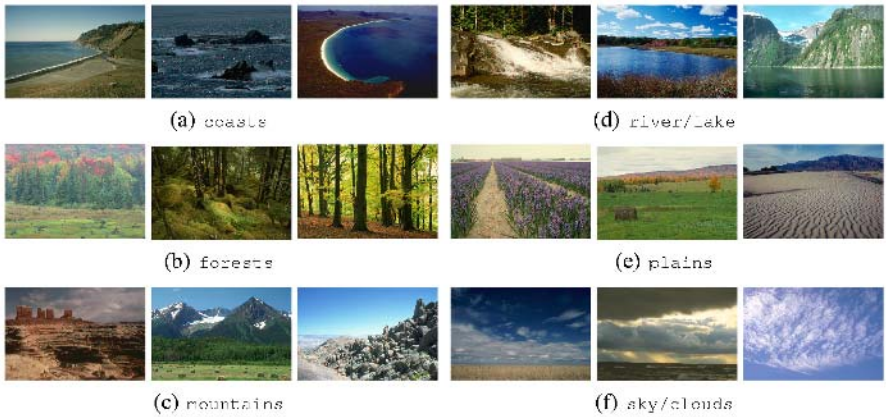
**Abstract.** In this paper, we present an approach for the retrieval of natural scenes based on a semantic modeling step. Semantic modeling stands for the classification of local image regions into semantic classes such as *grass*, *rocks* or *foliage* and the subsequent summary of this information in so-called concept-occurrence vectors. Using this semantic representation, images from the scene categories *coasts*, *rivers/lakes*, *forests*, *plains*, *mountains* and *sky/clouds* are retrieved. We compare two implementations of the method quantitatively on a visually diverse database of natural scenes. In addition, the semantic modeling approach is compared to retrieval based on low-level features computed directly on the image. The experiments show that semantic modeling leads in fact to better retrieval performance.

## 1 Introduction

Semantic understanding of images remains an important research challenge for the image and video retrieval community. Some even argue that there is an “urgent need” to gain access to the content of still images [1]. The reason is that techniques for organizing, indexing and retrieving digital image data are lagging behind the exponential growth of the amount of this data (for a review see [2]). Natural scene categorization is an intermediate step to close the semantic gap between the image understanding of the user and the computer. In this context, scene categorization refers to the task to group arbitrary images into semantic categories such as *mountains* or *coasts*.

First steps in scene category retrieval were made by Gorkani and Picard [3] (city vs. landscape), Szummer and Picard [4] (indoor/outdoor) and Vailaya et al. [5] (indoor/outdoor, city/landscape, sunset/mountain/forest). All these approaches have in common that they only use global information rather than local information. More recent approaches try to automatically annotate local semantic regions in images [6]-[9] but the majority does not attach a global label to the retrieved images. Oliva and Torralba find global descriptions for images based on local and global features but without an intermediate annotation step [10].

The general goal of our work is to find semantic models of outdoor scenes. In the context of image retrieval it reduces the amount of potentially relevant images. But it also allows to adaptively search for semantic image content inside a particular category (e.g. an image from the *mountains*-category, but with large forest, no rocks). Thus a



**Fig. 1.** Exemplary images for each category.

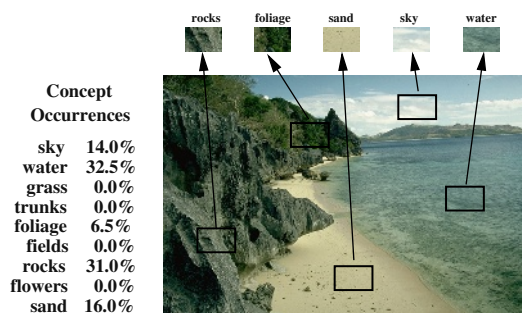
bottom-up step, i.e. scene categorization, and a top-down step, i.e. the use of category information to model relevant images in more detail, can be combined. For that goal, we employ a semantic modeling step. Semantic modeling stands for the classification of image regions into concept classes such as *rocks*, *water* or *sand* and the scene retrieval based on this information. The advantage of an intermediate semantic modeling step is that the system can easily be extended to more categories. Also, for local semantic concepts, it is much easier to obtain ground-truth than for entire images that are often ambiguous. In this paper, we compare two implementations of the semantic modeling approach for natural scene retrieval. In addition, we evaluate how the semantic modeling approach compares with direct low-level feature extraction.

Concerning the database, we paid special attention to using highly varying scenes. The database contains hardly two visually similar images. All experiments have been fully cross-validated in order to average out the fact that in such diverse databases certain test sets perform better than others. The goal is to find out how much profit semantic modeling brings in a realistic setting.

The paper is organized as follows. In the next section, our scene database and the image representation are discussed in detail. Section 3 explains the interplay between the semantic modeling step and the retrieval stage. Finally, Section 4 is devoted to several experiments that compare two different implementations of the system and quantify the performance of the semantic modeling approach vs. a low-level feature-based approach.

## 2 Natural Scene Categories

For the scene retrieval, we selected six natural scene categories: coasts, forests, rivers/lakes, plains, mountains and sky/clouds. Exemplary images for each category are displayed in Figure 1. The selected categories are an extension of the natural basic level categories of Tversky and Hemenway [11]. In addition, the choice of suitable categories has been influenced by the work of Rogowitz et al. [12].



**Fig. 2.** Semantic modeling

Obviously, these scene categories are visually very diverse. Even for humans the labeling task is non-trivial. Nonetheless, pictures of the same category share common local content, such as for example the local semantic concepts *rocks* or *foliage*. For example, pictures in the *plains*-category contain mainly *grass*, *field* and *flowers*, whereas *mountains*-pictures contain much *foliage* and *rocks*, but also *grass*. Based on this observation, our approach to scene retrieval is to use this *local semantic* information.

## 2.1 Concept Occurrence Vectors

By analyzing the local similarities and dissimilarities of the scene categories, we identified nine discriminant local semantic concepts: *sky*, *water*, *grass*, *trunks*, *foliage*, *field*, *rocks*, *flowers* and *sand*. In order to avoid a potentially faulty segmentation step, the scene images were divided into an even grid of 10x10 local regions, each comprising 1% of the image area. Through so-called concept classifiers, the local regions are classified into one of the nine concept classes. Each image is represented by a concept occurrence vector (COV) which tabulates the frequency of occurrence of each local semantic concept (see Figure 2). A more detailed image representation can be achieved if multiple COVs are determined on non-overlapping image areas (e.g. top/middle/bottom) and concatenated.

## 2.2 Database

Our database consists of 700 natural scenes: 143 coasts, 114 rivers/lakes, 103 forests, 128 plains, 178 mountains and 34 sky/clouds. Images are present both in landscape and in portrait format. In order to obtain ground-truth for the concept classifications, all 70'000 local regions (700 images \* 100 subregions) have been annotated manually with the above mentioned semantic concepts. Again, a realistic setting was of prime interest. For that reason, each annotated local region was allowed to contain a small amount (at maximum 25%) of a second concept. Imagine a branch that looms into the sky, but does not fill a full subregion (*sky* with some *trunks*) or a lake that borders on the forest (*water* with *foliage*). Due to these quantization issues, only 59'582 out of the 70'000 original annotated regions can be used for the concept classifier training since only those contain the particular concept with at least 75%. The rest has been annotated

**Table 1.** Confusion matrix of the local concept classification (k-NN classifier)

		Classifications in %									#regions
		<i>sky</i>	<i>water</i>	<i>grass</i>	<i>trunks</i>	<i>foliage</i>	<i>field</i>	<i>rocks</i>	<i>flowers</i>	<i>sand</i>	
True class	<i>sky</i>	<b>91.8</b>	5.7	0.0	0.1	0.5	0.2	1.6	0.0	0.2	15360
	<i>water</i>	9.5	<b>68.1</b>	2.4	0.0	6.0	3.8	9.0	0.1	1.2	7309
	<i>grass</i>	0.9	6.4	<b>34.4</b>	0.5	43.1	9.0	4.5	0.9	0.5	3541
	<i>trunks</i>	0.8	0.8	1.5	<b>28.0</b>	45.6	5.9	16.3	1.1	0.0	1516
	<i>foliage</i>	0.5	1.0	2.5	1.0	<b>85.1</b>	1.2	7.3	1.4	0.0	13470
	<i>field</i>	1.2	7.4	6.4	1.3	18.8	<b>34.8</b>	27.4	1.8	0.9	4070
	<i>rocks</i>	1.7	3.5	0.7	1.0	24.6	6.6	<b>61.0</b>	0.4	0.6	10567
	<i>flowers</i>	0.9	0.7	2.2	0.3	53.0	2.4	4.7	<b>35.5</b>	0.4	2051
	<i>sand</i>	6.3	19.7	6.3	0.4	2.2	16.5	32.6	0.3	<b>16.8</b>	1773

doubly. As some concepts exist in nearly all images and some only in a few images, the size of the nine classes varies between 1’516 (*trunks*) and 15’405 (*sky*) regions.

3 Two-Stage Scene Retrieval

In order to implement the semantic modeling step, the natural scene retrieval proceeds in two stages. In the first stage, the local image regions are classified into one of the nine concept classes. In the second stage, the concept occurrence vector is determined and the images are retrieved based on that concept occurrence vector. The following describes those two stages in more detail.

3.1 Stage I: Concept Classification

The local image regions are represented by a combination of a color and a texture feature. The color feature is a 84-bin HSI color histogram (H=36 bins, S=32 bins, I=16 bins), and the texture feature is a 72-bin edge-direction histogram. Tests with other features, such as RGB color histograms, texture features of the gray-level co-occurrence matrix, or FFT texture features, resulted in lower classification performance. The classification has been tested with both a k-Nearest-Neighbor ( $k = 30$ ) (k-NN) and a Support Vector Machine ( $C = 8, \gamma = 0.5$ ) (SVM) [13] concept classifier.

With 68.9% classification rate, the k-NN concept classifier showed a slightly inferior performance than the SVM concept classifier with 69.9% classification rate. Nevertheless, its resulting classifications perform better in the subsequent retrieval stage and will therefore be employed in all following experiments. The reason for this behavior is that the global classification rate usually improves to the benefit of the large classes (*sky, foliage*) and at the expense of the smaller classes (*field, flowers, sand*). Since these smaller classes are essential for scene retrieval, the overall classification accuracy on the first stage is not the most important performance measure.

The experiments have been performed with 10-fold cross-validation on *image* level. That is, regions from the same image are either in the test or in the training set but never in different sets. This is important since image regions from the same semantic concept

tend to be more similar to other (for example neighboring) regions in the same image than to regions in other images. The confusion matrix of the experiments with the k-NN concept classifier is depicted in Table 1. The confusion matrix shows a strong correlation between class size and classification result. In addition, we observe confusions between similar semantic classes, such as *grass* and *foliage*, *trunks* and *foliage*, or *field* and *rocks*.

The trained concept classifier is used to classify all regions of an image into one of the semantic classes. The experience showed that doubly annotated regions (e.g. with *sky* and *rocks* at the border between the sky and a mountain) were usually classified as one of those two semantic concepts.

### 3.2 Stage II: Scene Retrieval Based on Concept Occurrence Vectors

The output of the first stage is localized semantic information about the image. It specifies where in the image there are e.g. *sky* or *foliage*-regions and how much of the image is covered with e.g. *water*. From that semantic information, the concept occurrence vectors are determined. Experiments have shown that the retrieval performance improves if multiple concept occurrence vectors are computed either on three (top/middle/bottom) or five image areas. This leads to a resulting concept occurrence vectors of either length=27 or length=45.

In the following we propose two different implementations to semantically categorize images based on the concept occurrence vectors, namely a Prototype approach and an SVM approach. In the experiments those two implementations are compared and analyzed.

**Prototype approach to scene retrieval.** The prototype for a category is the mean over all concept occurrence vectors of the respective category members. Thus, the prototype can be seen as the most typical image representation for a particular scene category where the respective image does not necessarily exist. The bins or attributes of the prototype hold the information which amount of a certain concept an image of a particular scene category typically contains. For example, a forest-image usually does not contain any *sand*. Therefore, “*sand*-bin” of the forest-prototype is close to zero.

When determining the category of an unseen image, the Euclidean or the Mahalanobis distance between the image’s concept occurrence vector and the prototype is computed. The smaller the distance, the more likely it is that the image belongs to the respective category. By varying the accepted distance to the prototype, precision and recall for the retrieval of a particular scene category can be influenced.

**SVM approach to scene retrieval.** For the SVM-based retrieval of natural scenes we employ the LIBSVM package of Chen and Lin [13]. A Support Vector Machine is trained for each scene category. The input to the SVM are the concept occurrence vectors of the relevant images. The margin, that is the distance of an unseen concept occurrence vector to the separating hyperplane, is a measure of confidence for the category membership of the respective image. By varying the acceptance threshold for the margin, precision and recall of the scene categories can be controlled.

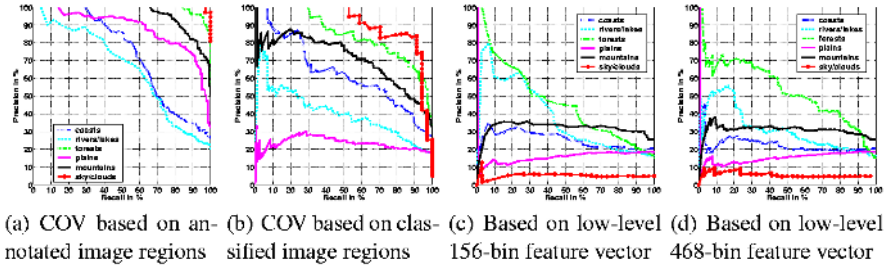


Fig. 3. Scene retrieval with Prototype approach

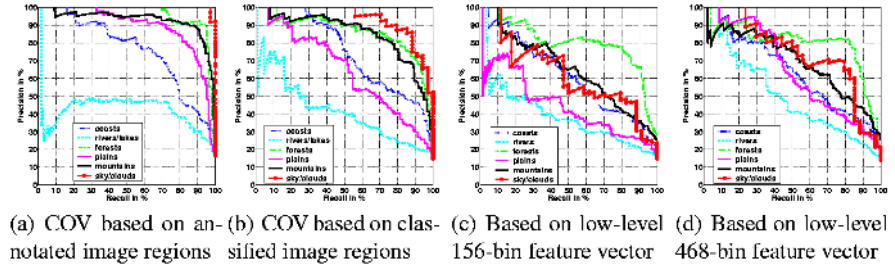


Fig. 4. Scene retrieval with SVM approach

## 4 Scene Retrieval: Experiments

Using the database described in Section 2.2, we conducted a set of experiments in order to compare the performance of the two retrieval implementations. In addition, it is evaluated whether the semantic modeling approach is superior to using low-level features of the images directly for retrieval. Performance measures are precision (percentage of retrieved images that are also relevant) and recall (percentage of relevant images that are retrieved). The precision-recall curves of the experiments are depicted in Fig. 3 for the Prototype approach and Fig. 4 for the SVM approach. Tables 2 and 3 summarize the Equal Error Rates (EER) of the experiments. Both concept classification and scene retrieval experiments are 10-fold cross-validated on the same ten test and trainings sets. That is, a particular trainings set is used to train the concept classifier, the SVM and the prototypes. Classification and retrieval are evaluated on the corresponding test set.

**Retrieval based on annotated image regions.** In the first experiment, we compared the performance of the Prototype vs. the SVM approach based on annotated patches. The goal of the experiment is to evaluate if the semantic modeling approach is effective given perfect data.

The results of the experiment are depicted in Fig. 3 (a) and Fig. 4 (a). The SVM approach outperforms the Prototype approach in 4 of 6 cases (Tables 2 and 3). Obviously, coasts and rivers/lakes are the most difficult categories. In fact, the detailed analysis



**Table 2.** Equal Error Rates for Prototype approach

Retrieval based on	coasts	rivers lakes	forests	plains	mountains	sky clouds
annotated regions	64.3%	61.9%	95.1%	79.7%	86.0%	97.1%
classified regions	57.3%	43.0%	74.8%	28.9%	66.9%	85.2%
156-bin feature vec.	29.4%	41.8%	45.6%	11.6%	33.8%	2.9%
468-bin feature vec.	25.7%	32.2%	50.5%	11.2%	32.5%	8.8%

**Table 3.** Equal Error Rates for SVM approach

Retrieval based on	coasts	rivers lakes	forests	plains	mountains	sky clouds
annotated regions	70.5%	47.4%	91.4%	81.3%	89.3%	97.2%
classified regions	61.0%	42.1%	80.6%	54.7%	78.1%	85.3%
156-bin feature vec.	56.6%	40.3%	77.6%	46.1%	59.0%	52.9%
468-bin feature vec.	57.3%	47.3%	81.4%	54.6%	63.8%	70.5%

of the retrieval results of those two categories shows that they are frequently confused. The main reason is that these two categories are in fact quite ambiguous. Even for the human annotator it is not clear into which category to sort a certain image that contains some water. It is especially those ambiguous images that are also wrongly retrieved by the retrieval system.

The SVM implementation has difficulties in modeling the rivers/lakes-category for small recall values since this category is not compact in the COV space. All other categories, that is plains, mountains, forests and sky/clouds, are retrieved with good to very good accuracy. Again the analysis of the retrieval results show that wrongly retrieved images are often semantically closer to the category that has been requested than to the “correct” category.

**Retrieval based on classified image regions.** In the next experiment, images with automatically classified local regions were considered. The concept classifier described in Section 3.1 and Table 1 was employed for the Stage I classification. Based on these classifications, the concept occurrence vector is determined. The retrieval result is depicted in Fig. 3 (b) and Fig. 4 (b). Here, the SVM approach again outperforms the Prototype approach in 5 of 6 cases (Tables 2 and 3). sky/clouds, mountains and forests have been retrieved especially well by the SVM. The loss compared to the annotated scenes is quite low. Compared to the retrieval in the annotated case, coasts are retrieved reasonably well.

The Prototype approach fails completely to retrieve plains, whereas the SVM is able to achieve an EER of 54.7%. The reasons for the general worse performance in the plains-category are the confusions of the concept classification stage. The plains-category can be discriminated by the detection of *field*, *grass* and *flowers*. These three concepts are confused to a large percentage with *rocks* and *foliage* (refer to Table 1). These strong mis-classifications lead to the observed low retrieval performance.

**Retrieval without semantic modeling step.** The last two experiments were carried out in order to find out whether the semantic modeling step is in fact beneficial for the retrieval task. Therefore this section will describe an experiment where we compare the retrieval results based on the concept occurrence vector vs. the performance using the low-level features directly as image representation. The same features as for the concept classification were used for the image representation: a concatenation of a 84-bin linear HSI color histogram and a 72-bin edge direction histogram. These features were once computed directly on the image, resulting in a global feature vector of length 156, and once on three image areas (top/middle/bottom), resulting in a feature vector of length  $3 \times 156 = 468$ . The “Prototype” approach now refers to the learning of a mean vector per category and the computation of the Euclidean distance between the mean vector and the feature vector of a new image. The results of these experiments are depicted in Fig. 3 (c)-(d) for the “Prototype” approach and Fig. 4 (c)-(d) for the SVM method.

Both the figures and the EERs in Table 2 clearly show that the “Prototype” approach based on low-level features fails compared to the semantic modeling based approach both for one image area and for three image areas. Probably the feature space is too high-dimensional and too sparse. For that reason also the introduction of more localized information through the use of three image areas does not bring any improvement compared to one image area.

In contrast, the low-level feature-based SVM approach performs surprisingly well compared to the SVM based on the semantic modeling step. The introduction of localized information by using three image areas also leads to a performance increase. The variation of the EER in the three-area feature-based approach is smaller than the approach based on the COV. Categories such as *sky/clouds* or *mountains* are not retrieved as good as with the semantic modeling approach and categories such as *rivers/lakes* are retrieved better than with the semantic modeling approach. But in summary, the performance increase in the *rivers/lakes*-category does not counter-balance the performance decrease in the *sky/clouds*- and *mountains*-category.

**Discussion.** Summarizing, we can draw two conclusions from the experiments. Firstly, the SVM implementation of the retrieval system outperforms the Prototype approach. Only single categories are retrieved better when using prototypes. Here, a combination of both methods might be advantageous.

Secondly, the semantic modeling step and an approach such as the concept occurrence vectors is beneficial for the retrieval of natural scene categories considered in this paper. For most categories, the EER obtained with the semantic modeling step is equal to or better than without the semantic modeling. Many of the wrongly retrieved images are in fact content-wise on the borderline between two categories. For that reason quantitative retrieval performance should not be the only performance measure for the semantic retrieval task. Still, the performance of the problematic categories *rivers/lakes* and *plains* can be improved by better concept classifiers in order to retrieve discriminant concepts with high confidence or better category models. One might, for example, employ different numbers of discriminant concepts and/or image areas per category in order to differentiate between *rivers/lakes* and *coasts*.

## 5 Conclusion

In this work, we presented an approach to natural scene retrieval that is based on a semantic modeling step. This step generates a so-called concept-occurrence vector that models the distribution of local semantic concepts in the image. Based on this representation, scene categories are retrieved. We have shown quantitatively that Support Vector Machines in most cases perform better than the retrieval based on category prototypes. We have also demonstrated that the semantic modeling step is superior to retrieval based on low-level features computed directly on the image. In addition, since ground-truth is more easily available for local semantic concepts than for full images, the system based on semantic modeling is more easily extendable to more scene categories and also to more local concepts. Further advantages of the semantic modeling are the data reduction due to the use of concept occurrence vectors and the fact that the local semantic concepts can be used as descriptive vocabulary in a subsequent relevance feedback step.

**Acknowledgments.** This work is part of the CogVis project, funded by the Comission of the European Union (IST-2000-29375) and the Swiss Federal Office for Education and Science (BBW 00.0617).

## References

1. Sebe, N., Lew, M., Zhou, X., Huang, T., Bakker, E.: The state of the art in image and video retrieval. In: Conf. Image and Video Retrieval CIVR, Urbana-Champaign, IL, USA (2003)
2. Smeulders, A., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. *IEEE Trans. on PAMI* **22** (2000)
3. Gorkani, M., Picard, R.: Texture orientation for sorting photos at a glance. In: Int. Conf. on Pattern Recognition ICPR, Jerusalem, Israel (1994)
4. Szummer, M., Picard, R.: Indoor-outdoor image classification. In: IEEE Int. Workshop on Content-based Access of Image and Video Databases, Bombay, India (1998)
5. Vailaya, A., Figueiredo, M., Jain, A., Zhang, H.: Image classification for content-based indexing. *IEEE Trans. on Image Processing* **10** (2001)
6. Maron, O., Ratan, A.L.: Multiple-instance learning for natural scene classification. In: Int. Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA (1998)
7. Town, C., Sinclair, D.: Content based image retrieval using semantic visual categories. Technical Report 2000.14, AT&T Laboratories Cambridge (2000)
8. Naphade, M., Huang, T.: A probabilistic framework for semantic video indexing, filtering, and retrieval. *IEEE Trans. on Multimedia* **3** (2001)
9. Barnard, K., Duygulu, P., de Freitas, N., Forsyth, D.: Object recognition as machine translation - part 2: Exploiting image data-base clustering models. In: European Conf. on Computer Vision, Copenhagen, Denmark (2002)
10. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. Journal of Computer Vision* **42** (2001)
11. Tversky, B., Hemenway, K.: Categories of environmental scenes. *Cogn. Psychology* **15** (1983)
12. Rogowitz, B., Frese, T., Smith, J., Bouman, C., Kalin, E.: Perceptual image similarity experiments. In: SPIE Conf. Human Vision and Electronic Imaging, San Jose, California (1998)
13. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. (2001) Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

# Unsupervised Text Segmentation Using Color and Wavelet Features

Julinda Gllavata<sup>1</sup>, Ralph Ewerth<sup>1</sup>, Teuta Stefi<sup>2</sup>, and Bernd Freisleben<sup>1,2</sup>

<sup>1</sup> ISFB/FK 615, University of Siegen, D-57068 Siegen, Germany  
{gllavata, ewerth}@fk615.uni-siegen.de

<sup>2</sup> Dept. of Math. and Computer Science, University of Marburg, D-35032 Marburg, Germany  
{stefi, freisleb}@informatik.uni-marburg.de

**Abstract.** Since the number of digital multimedia libraries is growing rapidly, the need to efficiently index, browse and retrieve this information is also increased. In this context, text appearing in images represents an important entity for indexing and retrieval purposes. Often, text is superimposed over complex image background and its recognition by a commercial optical character recognition (OCR) engine is difficult. Thus, there is the need for a text segmentation process, including background removal and binarization, in order to achieve a satisfactory recognition rate by OCR. In this paper, an unsupervised learning method for text segmentation in images with complex backgrounds is presented. First, the color of the text and background is determined based on a color quantizer. Then, the pixel color and the standard deviation of the wavelet transformed image are used to distinguish between text and non-text pixels. To classify pixels into text and background, a slightly modified k-means algorithm is applied which is used to produce a binarized text image. The segmentation result is fed into a commercial OCR software to investigate the segmentation quality. The performance of our approach is demonstrated by presenting experimental results for a set of video frames.

## 1 Introduction

Content-based image and video retrieval has attracted a lot of attention in recent years. Several approaches have been developed for indexing, querying and retrieving multimedia information. One possibility is to use the text embedded in images and video frames. Such kind of text offers important information for image and video understanding and is a very good entity for keyword-based queries.

In general, text appearing in images can be classified into two groups: scene text and artificial text [7]. Scene text is part of the image and does not represent any information about the image content, (traffic signs in an outdoor scene, etc.), whereas artificial text is laid over the image in a later stage (e.g. the name of somebody during an interview). Artificial text is usually a good key to index images or videos. To obtain such useful indexing data, an Optical Character Recognition (OCR) system must be used. If the original images are directly fed into an OCR system, the OCR results are often not sufficient due to the complexity of the image backgrounds. Text segmentation is aimed at simplifying or removing the background and thus increasing the text quality in order to achieve good results with OCR systems.

In this paper, the problem of text segmentation is addressed and an unsupervised learning method for text segmentation in images with complex background is presented. The problem of locating text in an image is not discussed here; our approach to this problem is presented in [3, 4]. The proposed text segmentation approach works as follows. First, the possible color of the text and background is determined using an appropriate vector color quantizer. Then, the pixel color and the standard deviation of the wavelet coefficients are used to distinguish between text and non-text pixels. A slightly modified k-means algorithm is used to classify text pixels in the image. This classification information is used to produce a binarized text image with black text on white background. This text image is then passed to an OCR system. We have used a commercial standard OCR software system to investigate the impact of our segmentation approach to recognition performance. The very good performance of our approach is demonstrated by presenting experimental results for a set of images.

The paper is organized as follows. Section 2 gives a brief overview of related work in the field. Section 3 presents the individual steps of our approach to text segmentation in detail. Section 4 describes the experimental results obtained for a set of images. Section 5 concludes the paper and outlines areas for future research.

## 2 Related Work

Agnihotri and Dimitrova [1] use the average of pixel values of the text image as a threshold for the binarization step. The authors assume that the average of pixel contours of the text box is closer to the average of the pixels marked as background on the text image. The problem of text embedded in complex background is not addressed, and performance results for segmentation or recognition are not reported.

Antani et al. in [2] have presented a simple text segmentation method. To agree with the possible polarity of text in an image, two segmented text regions are generated. A connected-component method is applied to the segmented result to remove the components that do not fulfill the specified aspect ratios. Finally, a score is assigned to each of the segmented images based on their text-like characteristics. The image with the highest score is selected as the input for an OCR system. No performance results for segmentation or recognition are presented.

Wolf et al. [16] have proposed a text localization, enhancement and binarization method for multimedia documents. The detected text boxes in multiple consecutive frames are used to create a high resolution text box using bilinear interpolation. Then, a combination of the classic thresholding algorithm presented in Niblack [10] and Sauvola [14] is used. During the binarization, a local threshold is calculated for each block separately. The method is tested on 60000 frames of different MPEG videos. The authors report to have achieved a character recognition rate of 85%.

In an approach proposed by Lienhart and Wernicke [7], the possible text and background color is estimated first. A 4(8)-neighborhood seed filling algorithm is applied to each text (background) pixel separately. Components that do not fulfill certain geometrical restrictions are removed. A binarization process follows where the global threshold is calculated as the mean of the text and background color. The threshold procedure is

applied differently for inverted and normal text. A recognition performance of 69.9% is given. The video test set used contains credits, commercial and news sequences.

Wu et al. [17] use a low pass Gaussian filter to smooth the image and compute an intensity histogram. Then, the histogram is also smoothed and the first peak from the left on the smoothed histogram is used as a threshold for the binarization process. The algorithm was tested on a set of photographs, newspapers, advertisements, personal checks (with 300 dpi). A character recognition rate of 84% is reported.

Loprestie and Zhou [8] have integrated the process of text segmentation into the recognition process. Two different methods are proposed for this purpose. The first one uses a polynomial surface fitting algorithm to recognize the characters. The second method is based on a fuzzy n-tuple classifier. Their methods are tested on a set of web pages. For the first method, a recognition rate of 69.7%, and for the second method, a recognition rate of 89.3% is reported. The two methods are trained with half of the test set.

Sato et al. [13] employ a sub-pixel interpolation method and a multi-frame integration schema to enhance the text image. Then, the extraction of characters is done through the combination of four filters. At the end, the image is binarized using a global threshold. The recognition rate using their own OCR is 83.5% for a CNN headline news test set.

Li et al. [6] first enhance the image resolution using Shannon up-sampling. Then, an adaptive threshold is used for binarizing the image. A block is marked as background only if its standard deviation is smaller than a fixed threshold. The recognition rate achieved for the test set used (images with low resolution) is 67.8%.

Odobez and Chen [11] have presented a multi-hypotheses approach based on a Markov random field (MRF) and on grayscale consistency constraints for text segmentation. The grey level distribution in text images is modeled as a mixture of Gaussians distributions. The assignment of each pixel to one of the Gaussian layers is based on prior contextual information, which is modeled by a MRF. Each layer is considered as a binary text image and is fed to the OCR as one segmentation hypothesis. The text image which gives the best recognition performance is considered as the output of the system. Finally, a simple evaluation method is applied to estimate the results of the OCR. The authors have reported a recognition rate of 93%. In their experiments, frames extracted from sports videos were used.

Hua et al. [5] have proposed a multiple frame text extraction schema. The frames where the same text appears in a clearly recognizable manner are averaged with each other to get a so called "man-made" frame. A block-based adaptive thresholding procedure applied to this frame concludes the segmentation process. They have reported a character recognition rate of 78% for a test set of MTV sequences.

Miene et al. [9] have presented a segmentation approach which consists of a region-growing algorithm for color segmentation and a method for separating text from the background based on geometrical constraints of characters. A character recognition rate of 81% is reported for a test set of videos from broadcast news and magazines.

### 3 Unsupervised Text Segmentation

The approach proposed in this paper is designed to segment horizontally aligned text in images of arbitrary font, size and color. The system input is an image and the coordinates of the text bounding boxes in the image. The text bounding boxes can be automatically computed, e.g. by using a text localization algorithm [3, 4] (the description of this algorithm is beyond the scope of this paper). In contrast to many other approaches, the use of global thresholds [1, 6, 7, 13] or local thresholds [5, 10, 14, 16, 17] is avoided in our approach by applying unsupervised clustering. Furthermore, the feature vector not only consists of color information, but also of wavelet coefficients in order to consider local text-specific characteristics. The segmentation approach can be divided into four main steps, which are described in detail below:

1. Resolution Enhancement;
2. Text Color Estimation;
3. Feature Selection and Normalization: Color and Wavelet Coefficients
4. Classification of Pixels.

#### 3.1 Resolution Enhancement

Most commercial OCR systems perform best if the image resolution is at least about 300 dpi. Furthermore, the subsequent segmentation steps also perform better, if the superimposed text is not too small. Thus, in case that the input image has a lower resolution, it is rescaled up to 300 dpi by a cubic interpolation.

#### 3.2 Text Color Estimation

In this step, the dominating text color is estimated for each text box, following an approach suggested in [7]. First, the number of colors in the text box is reduced to the *nr\_color* most dominating colors using a color quantization method [18]. The number of these colors can be set as a parameter. Then, two color histograms are calculated: the histogram of *nr\_text\_rows* center rows in the text box and the histogram of *nr\_backgr\_rows* rows directly above and underneath of the text box. Finally, the difference histogram of those two histograms is calculated. The text and the background color are defined as the maximum and the minimum of this difference histogram.

#### 3.3 Feature Selection and Normalization: Color and Wavelet Coefficients

We have investigated several features in order to find the best ones to classify pixels as text or background. The basic feature vector consists of the red, green, and blue pixel color component, scaled to the range [0, 1]. Furthermore, a small sliding window (e.g. 3\*3 pixels) is moved over the text box to consider local image properties. This technique is motivated by two observations. First, characters usually have a unique texture. Second, the border of superimposed characters results in high-contrast edges. Consequently, we apply the wavelet transform to the image to consider these properties and pass them to the subsequent clustering algorithm. The standard deviation of wavelet coefficients in

the sliding window is expected to be low within a character's texture, but high at its boundary. Thus, the character boundaries are enhanced in the segmented image by using this feature.

The general purpose of the wavelet transform is to decompose a signal into subbands at various scales and frequencies. The wavelet transform can be implemented using filter banks consisting of high-pass and low-pass filters. The application to an image consists of a filtering process in horizontal direction and a subsequent filtering process in vertical direction. For example, when applying a 2-channel filter bank (L: low-pass filter, H: high-pass filter), four sub-bands are obtained after filtering: LL, HL, LH and HH. The three high-frequency sub-bands (HL, LH, HH) strengthen edges in horizontal, vertical or diagonal direction, respectively. The wavelet coefficients of these sub-bands are used as features in our approach. Since the text-to-background contrast is expected to be high in the grey-scale transformed image but not in each color channel, we decided to apply the wavelet transform to the grey-scaled version of the image. In our approach, we have chosen a 5/3 wavelet filter bank evaluated in [15]. The standard deviation of wavelet coefficients in a sliding window at position  $(x, y)$  is defined as follows:

$$stdev_{window}(x, y) = \sqrt{\frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (I(x+i, y+j) - mean_{window})^2}, \quad (1)$$

where  $I(x+i, y+j)$  is the wavelet coefficient at pixel position  $(x+i, y+j)$ , and  $mean_{block}(x, y)$  is defined as:

$$mean_{window}(x, y) = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(x+i, y+j) \quad (2)$$

Except for the color components which are normalized as described before, the feature vector components are normalized as follows. If the lower bound of the original feature range is smaller than 0, i.e.  $[-m, n]$ , then the range for the features is shifted by  $m$  to  $[0, m+n]$ . Then, for a given image, the maximum  $max$  is computed for this feature, and the corresponding component in each feature vector is divided by  $max$  to normalize it to the range  $[0, 1]$ .

### 3.4 Classification of Pixels

The k-means algorithm is a popular and well-known algorithm to partition data into  $k$  clusters. The number of clusters  $k$  is assumed to be known in advance. In our case, there are two clearly distinguishable classes of pixels: "text" and "background".

The "text" cluster is initialized with the feature vector  $f_{text}$  that has the minimum Euclidian distance to the ideal feature vector  $f$  representing the "text" cluster, while the "background" cluster is initialized with the feature vector  $f_{background}$  that has the maximum Euclidian distance to  $f$ . For the "text" ("background") cluster, the ideal feature vector includes the predefined "text" ("background") color and the normalized wavelet coefficients (respectively their standard variation in the sliding window), which are 1 (0) in the ideal (non-ideal) case. Then, the k-means algorithm is applied to obtain clusters



whose members have the minimum Euclidian distance to the respective cluster mean feature vector. Finally, a segmented text box is generated where the pixels of the cluster “text” are painted in black, while the other pixels are painted in white. Thus, we obtain a binary image output, where black text appears on white background.

### 4 Experimental Results and Discussion

We have tested our text segmentation algorithm on various types of images. The test set consists of 20 images and covers a wide variety of background complexity and text type. There are 205 words and 1404 characters in those 20 images in total.

To evaluate the performance of the proposed text segmentation algorithm, character recognition experiments have been conducted. The recognition rate is used as an objective measure of the algorithm’s segmentation performance. We have used a demo version of the commercial OCR software ABBYY FineReader 7.0 Professional for recognition purposes. After segmentation, the segmented binary text image was fed manually to the OCR.

The following parameters were used for the estimation of text and background color:  $nr\_color = 6$ ,  $nr\_text\_rows = 4$ ,  $nr\_backgr\_rows = 2*2$ . The sliding window size was set to  $3 \times 3 pixels$ . No assumptions were made about the resolution of the input images. The wavelet 5/3 filter bank evaluated in [15] was used with the low-pass filter coefficients  $(-0.176777, 0.353535, 1.06066, 0.353535, -0.176777)$  and the highpass filter coefficients  $(0.353535, -0.707107, 0.353535)$ .

To estimate the best feature vector as well as to test the impact of subsequent resolution enhancement to character recognition, a first experiment was conducted with the original image resolution (72 dpi). Several compositions of the feature vector were investigated in this first stage:

- 1. RGB Color Components;
- 2. RGB Color Components + Wavelet Coefficients;
- 3. RGB Color Components + Standard Deviation of Wavelet Coefficients;
- 4. RGB Color Components + Wavelet Coeff. + Standard Dev. of Wavelet Coeff.

**Table 1.** The recognition performance after text segmentation using various feature vectors and compared with the original OCR results.

Feature Vector	Char. recogn.	Word recogn.
OCR on Original Image (72 dpi)	49.1%	24.9%
Color (72 dpi)	65.5%	40.5%
Color + Wavelet Coef. (72 dpi)	66.6%	42.9%
Color + Wavelet + StdDev. Wavelet C. (72 dpi)	70.9%	43.4%
<b>Color + StdDev. Wavelet. Coef. (72 dpi)</b>	<b>70.7% (+21.6%)</b>	<b>46.8% (+21.9%)</b>
OCR on Original Image (300 dpi)	65.1%	34.1 %
<b>Color + StdDev. Wavelet Coef. (300 dpi)</b>	<b>85.9% (+20.8%)</b>	<b>65.4% (+31.3%)</b>

The OCR results are shown in Table 1. The best overall result considering both character and word recognition for the 72 dpi resolution images was achieved for the feature vector consisting of R, G and B color components and the standard deviation of wavelet coefficients within the sliding window (70.7% for character recognition and 46.8% for word recognition). Furthermore, we applied the text segmentation algorithm using this feature vector to a resolution enhanced image (up to about 300 dpi). As a result, after segmenting this high resolution image, the character recognition rate increased to 85.9% while 65.4% of the words were recognized correctly, as also shown in Table 1. Some examples of our text segmentation algorithm and the recognition results are presented in Figure 1.

We agree with Wolf et al. who remarked in [16] that a comparison with other approaches is very difficult, if not impossible, due to lack of a common test base. For example, in [11] as well as in [16] the classical binarization method of Otsu [12] was implemented and compared with the performance of the authors' own approach. In [16], the authors' implementation of Otsu's method led to a low character recognition rate of 47.3%, and their own approach achieved 85%. In [11], the implementation of Otsu's method obtained a character recognition rate of 88%, while the authors reported 93% for their own approach. Clearly, it is hard to conclude which approach performed better.

We believe that the reported performance results for our approach are at least competitive to the best results reported by other researchers. Although our test set is relatively small, it consists of various low resolution images which partially have very complex backgrounds and include various font types and sizes. Loprestie and Zhou [8] achieved a better result (89.3%), but during the recognition process similar characters (e.g. "c", "e") were considered as one class. Sato et al. [13] reported 83.5%, but only frames from CNN news videos were investigated containing only two different font types. Wolf et al. [16] achieved a character recognition rate of 85.4% for a test set of video frames containing 3519 characters. A comparison with [16] is difficult, too, since multiple consecutive frames were used in their approach to build a highresolution text box.

## 5 Conclusions

In this paper, we have presented an unsupervised algorithm for text segmentation in images with complex background. First, the text color is determined using a colorquantizer and line histograms. Then, the R, G, and B color components and the three high-frequency wavelet coefficients are used as the features for the subsequent classification into text and background pixels. The classification is done by a slightly modified k-means algorithm. Several possible feature vector compositions were investigated on a test set of images, consisting mostly of single video frames with complex backgrounds and different font types. The best results were achieved if the resolution of the original image was increased from 72 dpi up to 300 dpi and then a feature vector including the color components R, G, and B, and the standard deviation of wavelet coefficients within a small sliding window was used. In this test case, 85.6% of the characters and 65.4% of the words were recognized correctly. The word recognition is nearly two times higher than the word recognition on the original images (34.1%), while the character recognition is about 20% higher (65.1% on the original images).



**Fig. 1.** The text segmentation and recognition results: (a) The original images; (b) The result of our text segmentation and binarization algorithm; (c) The OCR results of both segmented images using ABBYY FineReader 7.0 Professional.

There are several areas for further research. The extension of the proposed text segmentation approach to videos instead of images will be considered in the future. Furthermore, the integration of a freely available OCR system will be investigated to support the whole processing chain from the input image to the ASCII-text at the end. The implementation of a complex system for automatic indexing of images and videos and their content-based retrieval will be also investigated.

**Acknowledgements.** This work is financially supported by the Deutsche Forschungsgemeinschaft (SFB/FK 615, Teilprojekt MT). The authors would like to thank M. Gollnick, M. Grauer, F. Mansouri, E. Papalilo, R. Sennert and J. Wagner for their valuable support.

## References

1. Agnihotri, L., Dimitrova, N.: Text Detection for Video Analysis. Proc. of International Conference on Multimedia Computing and Systems. Florence (1999) 109–113
2. Antani, S., Crandall, D., Kasturi, R.: Robust Extraction of Text in Video. Proc. of IEEE International Conference on Pattern Recognition, Vol. 1. Barcelona (2000) 1445–1449
3. Gillavata, J., Ewerth, R., Freisleben, B.: Finding Text in Images via Local Thresholding. Proc. of the 3rd IEEE Int'l Symposium on Signal Processing and Information Technology. Darmstadt, Germany (2003)
4. Gillavata, J., Ewerth, R., Freisleben, B.: A Robust Algorithm for Text Detection in Images. 3rd Int'l Symposium on Image and Signal Processing and Analysis. Rome (2003) 611–616
5. Hua, X. S., Yin, P., Zhang, H. J.: Efficient Video Text Recognition Using Multiple Frame Integration. Proc. of IEEE International Conference on Image Processing, Vol. 2. Rochester, New York (2002) 397–400

6. Li, H., Kia, O., Doermann, D.: Text Enhancement in Digital Videos. SPIE Vol. 3651: Document Recognition and Retrieval VI. (1999) 2–9
7. Lienhart, R., Wernicke, A.: Localizing and Segmenting Text in Images and Videos. IEEE Transact.on Circuits and Systems for Video Technology, Vol. 12, Nr. 4. (2002) 256–258
8. Loprestie, D., Zhou, J. Y.: Locating and Recognizing Text in WWW Images. Information Retrieval. Kluwer Academic Publishers. (2000) 177–206
9. Miene, A., Hermes, Th., Ioannidis, G.: Extracting Textual Information from Digital Videos. Proc. of IEEE Sixth International Conference on Document Analysis and Recognition. Seattle, Washington (2001) 1079–1083
10. Niblack, W.: An Introduction to Digital Processing. Prentice Hall (1986) 115–116
11. Odobez, J. M., Chen, D.: Robust Video Text Segmentation and Recognition with Multiple Hypotheses. Proc. of IEEE International Conference on Image Processing 2002, Vol. II. Rochester, New York (2002) 433–436
12. Otsu, N.: A Threshold Selection Method from Gray-Level Histograms. IEEE Transactions on Systems, Man and Cybernetics, 9 (1). (1979) 62–66
13. Sato, T., Kanade, T., Huges, E. K., Smith, M. A., Satoh, S.: Video OCR: Indexing Digital News Libraries by Recognition of Superimposed Caption. ACM Multimedia Systems, Vol. 7, No. 5. Orlando, Florida (1999) 385–395
14. Sauvola, J., Seppänen, T., Haapakoski, S., Pietikäinen, M.: Adaptive Document Binarization. Proc. of International Conference on Document Binarization, Vol. 1. (1997) 14–152
15. Villasenor, J., Belzer, B., Liao, J.: Wavelet Filter Evaluation for Efficient Image Compression. IEEE Transactions on Image Processing, Vol. 4. (1995) 1053–1060
16. Wolf, C., Jolion, J. M., Chassaing, F.: Text Localization, Enhancement and Binarization in Multimedia Documents. Proc. of International Conference on Pattern Recognition, Vol. 4. Quebec City, Canada (2002) 1037–1040
17. Wu, V., Manmatha, R., Riseman, E. M.: Textfinder: An Automatic System to Detect and Recognize Text in Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, Issue 11. (1999) 1224–1229
18. Wu, X.: YIQ Vector Quantization in a New Color Palette Architecture. IEEE Transactions on Image Processing, Vol. 5, No. 2. (1996) 321–329

# Universal and Personalized Access to Content via J2ME Terminals in the DYMAS System

Ana García<sup>2</sup>, José M. Martínez<sup>1</sup>, and Luis Herranz<sup>1</sup>

<sup>1</sup>Grupo de Tratamiento de Imágenes  
<http://gti.ii.uam.es>

Escuela Politécnica Superior, Universidad Autónoma de Madrid,  
Ctra. de Colmenar Viejo, Km 15, E-28049 Madrid, Spain  
[JoseM.Martinez@uam.es](mailto:JoseM.Martinez@uam.es)

<sup>2</sup>E.T.S.I.Telecomunicación, Universidad Politécnica de Madrid,  
Ciudad Universitaria s/n, E-28040 Madrid, Spain

**Abstract.** This paper presents the content adaptation application for mobile terminal within the current status of the Deferred Time Environment (DTE) of the DYMAS system. The DTE enables universal and personalized access to multimedia content over fixed and mobile networks. After introducing the system architecture, we show the differences between the functionalities for fixed and mobile networks, and detail the development of the application targeted to J2ME terminals. The system uses MPEG-21 for the description of sessions (terminal and network capabilities and user preferences) and MPEG-7 for content descriptions, which are the base for the Annotation, Search and Browsing, and Transcoding appliances.

## 1 Introduction

Universal Multimedia Access (UMA) [1] refers to the capability of access to rich multimedia content through any client terminal and any network. The development of new wireless networks, providing multimedia capabilities, and a wide and growing range of client terminals makes the adaptation of content an important issue in future mobile multimedia services.

Different authors have published about general issues and architectures for UMA systems [1][2][3][4][5], but there are not so many papers about prototypes, test-beds, or implementations (e.g., [6][7]). The DYMAS project provides, besides a real-time environment for generating metadata to enable the provision of added-value MHP applications synchronized with content[8], an environment enabling the provision of alternative multimedia services based on content broadcasted in digital television channels[9]. These services rely on the UMA concept and associated technologies and standards, with a special focus on MPEG-7 and MPEG-21 (in [6] the system uses only MPEG-21, whilst in [7] only MPEG-7 was used).

Section 2 introduces the DYMAS system; Section 3 presents the current architecture of the Deferred Time Environment (DTE) enabling UMA functionalities within DYMAS. Sections 4 and 5 describe, respectively, the different descriptions used and

their relation with the MPEG-7 and MPEG-21 specifications, and the functionalities of the J2ME mobile user applications. Section 6 concludes the paper.

## 2 Overview of the DYMAS System

Figure 1 depicts an overview of the DYMAS System architecture. It mainly describes a processing system with one information input (a DVB Transport Stream) and two information outputs (the modified DVB-TS and audiovisual services directed to other alternative access networks).

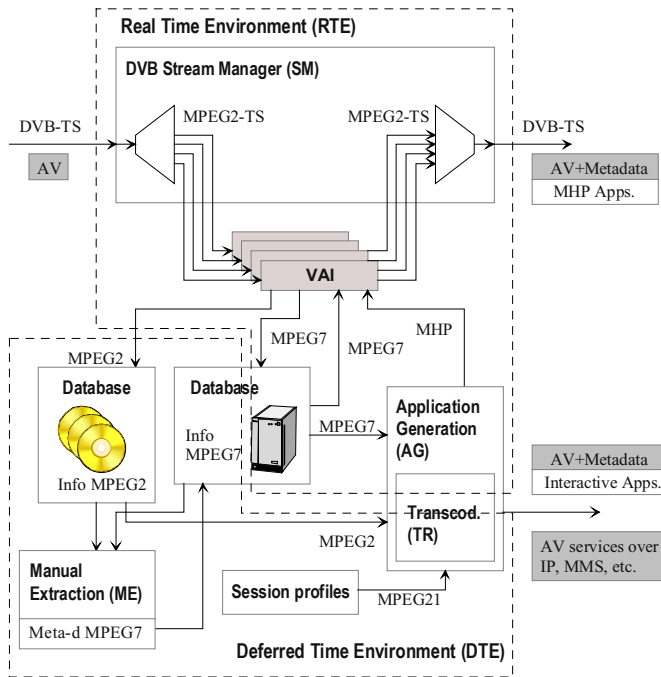


Fig. 1. Block diagram of the DYMAS System

The system relies on technology for automatic content extraction from audiovisual information, which is highly resource consuming, and just able to cope in real-time with low level basic features. These features are the basis for on-line service provision, that is, for the Real-Time Environment (RTE) [8]. Besides real-time added value for interactive TV applications, the DYMAS framework also considers the provision of Universal Multimedia Access services that do not have a real-time requirement, but can conversely be offered with some delay. This is a responsibility of the Deferred-Time Environment (DTE).

The DYMAS system uses the framework of MPEG-7[10], currently mainly the Multimedia Description Schemes [11] specification, to provide description metadata of the multimedia content. Some parts of the descriptions are generated in the RTE

and besides their use in interactive television applications, they are stored in the MPEG-7 database where descriptions are enriched via manual annotation and additional (non real time) automatic and supervised algorithms for feature extraction. These enhanced descriptions are the base for the UMA services provided by the DTE. Additionally the DTE makes use of MPEG-21[12], mainly the Digital Item Adaptation specification[13], to provide description metadata of the session (terminal and network capabilities and user preferences) in order to perform content adaptation.

### 3 Current DTE Architecture

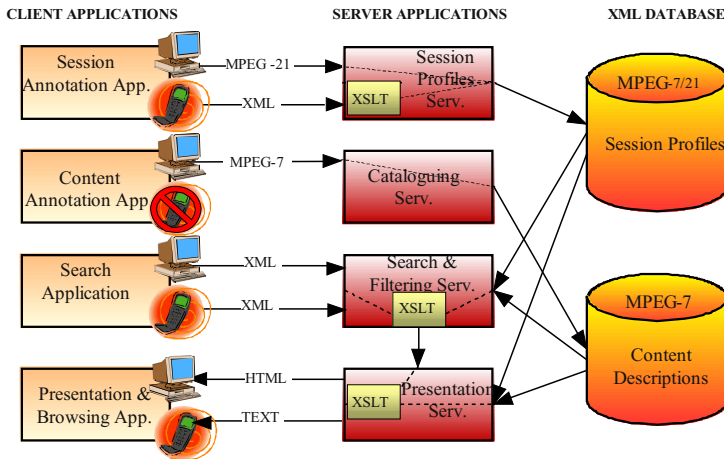
The current DTE provides universal and personalized access to the MPEG-2 database over the fixed and mobile networks. At the current status of the system implementation, the fixed applications run over Web Browsers running on standard PCs[14] and the mobile applications run over Java terminals[15] (support for MMS terminals will be also available soon[16]). Both fixed and mobile applications have the same functionalities although there are some differences because the wireless resources are scarce.

The DTE is composed by two main subsystems: the Metadata subsystem in charge of annotating, searching and browsing, and the Transcoding subsystem in charge of performing content adaptation to each particular session.

#### 3.1 Metadata Subsystem

The Metadata subsystem (see Figure 2) provides four client terminal specific applications (three in the case of the J2ME clients) that run over a common set of server applications and databases.

- The Annotation application provides an interface to edit XML descriptions using a set of MPEG-7 and MPEG-21 description tools. These descriptions can refer to the multimedia content (according to MPEG-7) and be used to catalogue new multimedia content, or can be descriptions of the session, including user preferences, terminal and network descriptions (according to MPEG-21 DIA, plus MPEG-21 Digital Item Declaration). There are currently two annotation applications. For fixed terminals (PC with web browsers) the annotation application allow content and session annotation[9], whilst for mobile terminals only session descriptions can be generated via the corresponding session annotation application (see section 5.1).
- The Search application provides an automatic search including (simultaneously or not depending on user selection) automatic filtering based on the user preferences (part of the session profile) and a query driven search. When the search results are available, the server sends the results to the client terminal in the appropriate format to be processed and presented correctly.



**Fig. 2.** Metadata subsystem

- The Browsing application allows users to select content among the obtained results and to access a detailed description of the content.

As we can see in Figure 2, the information sent to clients of fixed networks is a set of HTML pages, while mobile clients receive plain text that the client application interprets.

The XSL Transformations have a great importance in the applications architecture [9]. In fact, the search application could be understood as a set of chained transformations, from client request (in XML/MPEG-7 format) to server response (in HTML or plain text), including search, filtering and presentation processes. XSLT pattern matching allows the search engine to process the XML query in a natural way, transforming it in an output XML document with the results of the search. In the same transformation, the filtering is performed making use of the user profile description. At last, results are transformed one more time to format the results in order to send the response back to the user.

### 3.2 Transcoding Subsystem

The transcoding process modifies some media characteristics (width and high pixels number of the image, video bitrate, frame size, audio bitrate, sample rate, etc) of the MPEG video in the content database. The values of the selected parameters are obtained from the terminal capabilities and the network characteristics described in the MPEG-21 session profile.

The transcoding content can be directly sent to the client through streaming, as in the application for wired PCs, or it can be stored in the server to be downloaded later through a HTTP request by the client, as in the application for J2ME mobile terminals. As expected, both approaches have advantages and disadvantages (real time, delay, persistent copy for further reuse, ...).



## 4 Content, Session, and Query Descriptions

Within the system, we consider three descriptions: the content description, the session description, and the query description.

The content description uses MPEG-7, and is based on the MPEG-7 Simple Profile[17]. The Simple Profile is based on the MPEG-7 standard but there are some restricted descriptors. Content descriptions are created using the corresponding part of the annotation application of the fixed terminals. Although the mobile Java terminals don't have the possibility to make annotations of the content description, because they don't have the editor application, they use these descriptions to obtain information of the multimedia contents and to carry out the search application.

The session description uses MPEG-21 DIA (Digital Item Adaptation)[13] description tools. The session description is split into three subdescriptions (MPEG-21 compliant), depending on the context element to be considered. A session description contains links to a network description, a terminal description and a user description (see Fig. 3). Session descriptions are created using the corresponding part of the Annotation application (in a future, network and terminal will be detected automatically and user preferences updated automatically inspecting -if allowed- user's usage history). The user description informs about the client's content preferences and the client's preferences presentation. The terminal description contains information about the decoding, the display and the audio output capabilities, besides the power, the storage and the data IO characteristics. The network description contains data transmission characteristics, delay and error patterns, etc.

The query description uses a modified MPEG-7 profile which represents a partial content description that the user wants to match in the database. Therefore these "queries" are also XML descriptions similar to MPEG-7 content descriptions. However, as MPEG-7 is not designed for query, the query description is compliant with a modified MPEG-7 schema which includes some extra attributes (e.g., *case-sensitive*, *just-included*), and unconstrains some description elements.

## 5 J2ME Client Applications

The technology used in the development of the applications for mobile terminals has been J2ME (Java 2 Micro Edition, the specific Java platform for wireless devices), which allows the development of small programs called MIDlets. These small applications should be developed keeping in mind that in the wireless world the network data rates and processors are slow and memory is scarce. These constraints force that the client applications should be designed as small as possible.

The client applications for mobile terminals have been designed as elemental as possible to avoid giving to the user all the complexity of navigating through too many screens to obtain the desired contents.

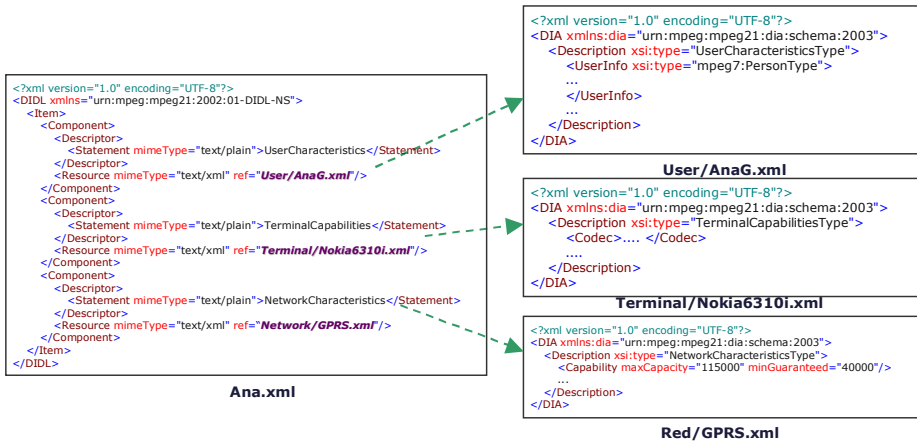


Fig. 3. Session description and subdescriptions

### 5.1 Session Annotation Application

The first thing the user has to do to be able to use the application is to register with a session profile. When the user is registered, he/she identifies his/her preferences, the terminal capabilities and the networks characteristics that he/she uses. With this information stored in the database, the application can obtain the values to configure the transcoding of the multimedia contents. If the user doesn't have a session profile assigned, he/she can create it with the session annotation application, which is available both for users of fixed networks and for users of mobile networks.

The creation of a session description consists on editing or selecting three different descriptions (user preferences, terminal capabilities and network characteristics) as it has been explained above. Selection of a created description, which are stored at the server, is done via a list. The user has the possibility of inspecting the descriptions to decide if it matches his/her desired session profile. To edit a description the user introduces the values of the fields and the application internally produces the XML description using a parser. When the description has been created, it is sent to the server for persistent storage. Each new description can be reused by future users.

### 5.2 Search and Browsing Applications

The search application allows the user to create an XML query description (see above) with the information introduced in the available fields, giving the option of importing the user preferences annotated in the session profile. Once the request has been sent to the server, it is processed and a set of results is sent back to the mobile device.

The client application receives the results of the search in plain text, interprets them and generates a list. When the user selects an element of the list, all the related meta-data available from the content description appears on the screen.

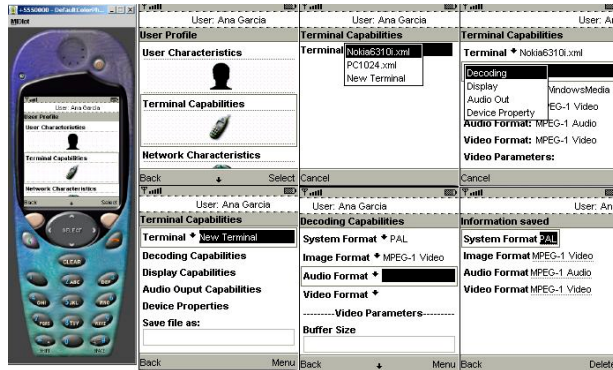


Fig. 4. Session annotation application

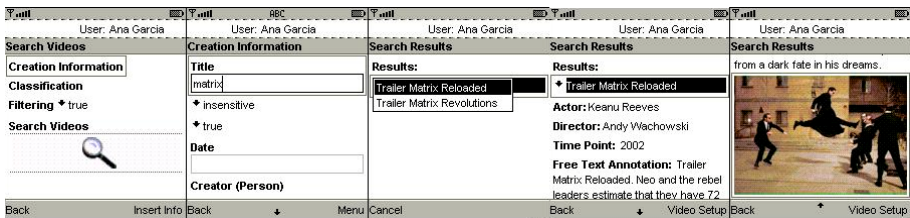


Fig. 5. Search and Browsing applications

### 5.3 Transcoding Application

Before doing the request of the selected video, all the parameters, which are used to configure the transcoding process, are displayed to the user. As it has been said previously, the values of the parameters have been obtained from the session profile. This session profile is the one that the user has configured when registering to the application. These values are not the definitive ones because the user can modify them inside a range of available values depending on particular circumstances of the running session. This range of values is usually limited by the network characteristics or the terminal capabilities.



Fig. 6. Transcoding Application: configure and play video

The last step to play the transcoded video is to send to the server the configuration parameters for the transcoding process, that will only be carried out if there is not an available copy in the variations cache database with the required media characteristics. After transcoding and storing the new variation in the cache database, if these steps were required, the application downloads the video to the terminal and the video begins to reproduce automatically. The player allows the user to carry out several basic operations with the video, for example, stop it, rewind it, put it to full screen, mute audio, etc.

## 6 Conclusions

As we can see, nowadays the users who want to get multimedia contents can use different types of networks and terminals. The main objective of UMA concept is to adapt these multimedia contents to the networks characteristics and the terminal capabilities. The current DYMAS system integrates both fixed access service with web browsers and wireless access with J2ME mobile terminals.

Although mobile terminals tend to have the same functionalities than fixed terminals, there are some differences because the network data rates and processors are slow and memory is scarce. Both services (fixed and mobile access) include a session annotation application to describe a session profile, and a search and browsing application. The content annotation application which allows cataloguing content based on the MPEG-7 standard is only available for the fixed access service. With the session annotation applications the user can describe his/her preferences, the terminal capabilities and the networks characteristics with an MPEG-21 description. This information configures the transcoding system when the search application finds the multimedia content that the user wishes.

Preliminary testing of the currently implemented and integrated applications (web for fixed PCs and J2ME applications) indicates that the MPEG-2 database can be accessed via alternative networks and services providing content adaptation to terminal characteristics and personalization to user preferences, that is, UMA functionalities. The main problem still remains the requirement for “timely” annotation of content for providing the personalization to user preferences. After final integration of the support for MMS terminal, testing and validation of the current version of the UMA functionalities will follow.

**Acknowledgements.** Work partially supported by the Comisión Interministerial de Ciencia y Tecnología of the Spanish Government under project TIC2002-03692 (DYMAS). Part of the work of Ana García was partially supported by Amena under a grant of the “Cátedra Amena” of the Universidad Politécnica de Madrid.

## References

- [1] A. Vetro, C. Christopoulos, T. Ebrahimi (guest editors), "Universal Multimedia Access (special issue)", *IEEE Signal Processing Magazine*, 20 (2), March 2003.
- [2] R. Mohan, J.R. Smith, C.-S. Li, "Adpating Multimedia Internet Content for Universal Access", *IEEE Transactions on Multimedia*, 1(1):104-114, March 1999.
- [3] J.R. Smith, "Universal Multimedia Access", in *Proc. SPIE Multimedia Systems and Applications IV*, vol. 4209, Nov. 2000.
- [4] A. Perkis, Y. Abdeljaoued, Ch. Christopolous, T. Ebrahimi, J. Chicharo, "Universal Multimedia Access from Wired and Wireless Systems", *Birkhauser Boston Transactions on Circuits, Systems and Signal Processing (special issue on Multimedia Communications)*, 20(3):387-402, 2001.
- [5] E. Fossbakk, P. Manzanares, J.L. Yago, A. Perkis, "An MPEG-21 framework for streaming media", in *Proc. of IEEE Multimedia Signal Processing 2001*, pp. 147-152, October 2001.
- [6] A. Perkis, J. Zhang, T. Halvorsen, J.O. Kjode, F. Rivas, "A streaming media engine using digital media adaptation", in *Proc. of IEEE Multimedia Signal Processing 2002*, pp. 73-76, December 2002.
- [7] J.M. Martínez, C. González, O. Fernández, C. García, J. de Ramón, "Towards Universal Access to Content using MPEG-7", *Proc. of ACM Multimedia 2002 Conference*, pp. 199-202.
- [8] J. Bescós, J.M. Martínez, N. García, "Real-time audiovisual feature extraction for on-line service provision over DVB streams", *Visual Content Processing and Representation-VLBV03, LNCS Vol. 2849*, Springer Verlag, 2003, pp. 141-148.
- [9] L. Herranz, José M. Martínez, "Towards Universal and Personalized Access to Audiovisual Content in the DYMAS System", *Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, April 2004.
- [10] J.M. Martínez, R. Koenen, F. Pereira, "MPEG-7: the generic multimedia content description standard", *IEEE Multimedia*, 9(2): 78-87, April-June 2002
- [11] MPEG MDS Group, "MPEG-7 Multimedia Content Description Interface, Part 5: Multimedia Description Schemes IS", ISO/IEC N4242, July 2001.
- [12] I. Burnett, R.van de Walle, K. Hill, J. Bormans, F. Pereria, "MPEG 21: Goals and Achievements", *IEEE Multimedia*, 10(4):60-70, Oct.-Dec 2003.
- [13] MPEG MDS Group, "MPEG-21 Multimedia Framework, Part 7: Digital Item Adaptation FCD", ISO/IEC N5845, July 2003.
- [14] L. Herranz, *Acceso Multimedia Universal mediante MPEG-7 y MPEG-21 a través de terminales y redes fijas (Universal Multimedia Access using MPEG-7 and MPEG-21 through fixed terminals and networks)*, Master Thesis, E.T.S.I.T, Universidad Politécnica de Madrid, July 2003.
- [15] A. García, *Acceso Multimedia Universal mediante MPEG-7 y MPEG-21 a través de terminales y redes móviles (Universal Multimedia Access using MPEG-7 and MPEG-21 through mobile terminals and networks)*, Master Thesis, E.T.S.I.T, Universidad Politécnica de Madrid, March 2004.
- [16] M. Padilla, *Sistema de transcodificación de vídeos para mensajería multimedia (Video transcoding system for multimedia messaging)*, Master Thesis, E.T.S.I.T, Universidad Politécnica de Madrid, March 2004.
- [17] MPEG Requirements Group, "MPEG-7 Profiles Under Consideration", ISO/IEC N6039, October 2003.

# Task-Based User Evaluation of Content-Based Image Database Browsing Systems

Timo Ojala<sup>1</sup>, Markus Koskela<sup>2</sup>, Esa Matinmikko<sup>3</sup>, Mika Rautiainen<sup>1</sup>,  
Jorma Laaksonen<sup>2</sup>, and Erkki Oja<sup>2</sup>

<sup>1</sup> MediaTeam Oulu, University of Oulu, Finland  
{timo.ojala, mika.rautiainen}@ee.oulu.fi  
<http://www.mediateam.oulu.fi>

<sup>2</sup> Laboratory of Computer and Information Science, Helsinki University of Technology,  
Finland  
{markus.koskela, jorma.laaksonen, erkki.oja}@hut.fi

<sup>3</sup> Mawell Ltd, Finland  
esa.matinmikko@iki.fi

**Abstract.** This paper presents a task-based user evaluation of two content-based image database browsing systems. The performance of the two systems is compared to that of a commercial image database management program, which does not employ content-based information. Experimental results show that content-based cues improve the efficiency of the browsing considerably. Guidelines for system design are derived from the user feedback.

## 1 Introduction

Literature proposes numerous methods for assessing the usability and performance of interactive systems such as image database search and browsing systems [10,12]: observation, think aloud, questionnaires, interviews, focus groups, logging actual use, user feedback, heuristic evaluation, pluralistic walk-through, formal usability inspection, empirical methods, cognitive walkthroughs, formal design analysis, etc. Despite several decades of retrieval experiments, the early quantitative measures of precision and recall are still the most widely adopted approaches. [1,4,13]

Qualitative user tests allow researchers to obtain knowledge how users perceive the system's performance and usability. The general nature of search tasks in visual information systems degrades the value of synthetic testing, such as [3], in real operational environments. However, the artificially generated performance numbers have an important role in the selection of technological alternatives for the system.

This paper presents task-based user evaluation of two content-based image browsing systems IIRO [11] and PicSOM [7]. The performance of the two systems is compared to that of the commercial ACDSee program [14]. ACDSee does not employ any content-based methods, but search and management of images is based on efficient browsing of thumbnail images.

## 2 Browsing Systems

### 2.1 IIRo

The IIRo system [11] is based on unsupervised clustering of content-based metadata using self-organizing maps (SOM, [5]). Information visualization techniques, multi-resolution object layers and a zoom view based on the focus+context technique [2] are employed to improve the user's interaction with the browsing system. The user is offered a simultaneous focused presentation of the selected object and other similar ones, while still maintaining a view to the entire database, which prevents potential straying of the user during browsing.

The multi-resolution index structure of the IIRo system is implemented with a self-organizing map, which is trained with the content-based metadata extracted from the objects of the database. The SOM provides topological ordering of the objects at the so-called root level, from which the browsing level is obtained by subsampling in both horizontal and vertical directions so that objects residing in the nodes within a given area are pooled into a single node in the browsing level. Subsampling is dynamic, producing a desired browser view.

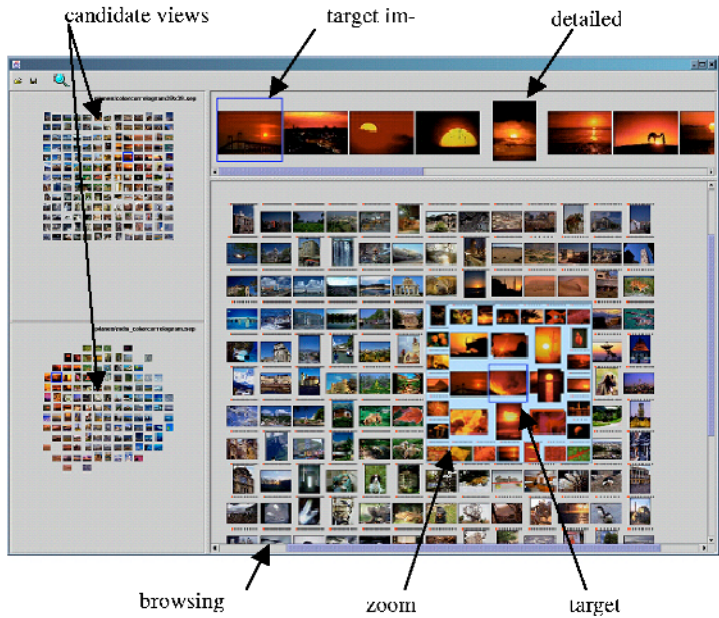


Fig. 1. The graphical user interface of the IIRo system.

IIRo's graphical user interface is illustrated in Fig. 1. The UI is a single window, which is divided horizontally into two parts. The left part contains miniature visuali-

zations of the candidate views into the database constructed with different metadata. In the right part are the actual browsing view and the detailed view showing the images in the current target node.

In the browsing view each node is visualized by one of the images residing in the node. The target image, the image visualizing the current target node (focus) is highlighted, and the browsing view is panned so that the target node is close to the center of the view. The target image is highlighted simultaneously in all visualizations according to the linking+brushing metaphor, to help the user comprehend the relationships between different visualizations.

The browsing view is used solely with a pointing device (mouse) so that the node of interest is selected with the right button. When a node is selected, one of the images residing in the node is visualized as the target image. If the current target node is re-selected, then a different image is chosen as the target image. The user is informed about the number of images in a single node with a row of dots above the image selected to visualize the node. By double-clicking the target image, the user can visualize it in full size in a separate window according to the details-on-demand metaphor.

The browsing view includes an optional zoom view based on the focus+context technique. With the left button of the mouse the user can fire up the zoom view, which provides a more detailed view of the images close to the current target image. To avoid confusing the user with the appearance of the zoom view, it is created as an animation, originating from the current target image and slowly expanding to its full size. The zoom view has a lighter background color to distinguish it from the browsing view. Individual images are visualized as thumbnails: 20x20 pixels in the panel of candidate views, 100x100 in the browsing view, and 150x150 in the center of the zoom view.

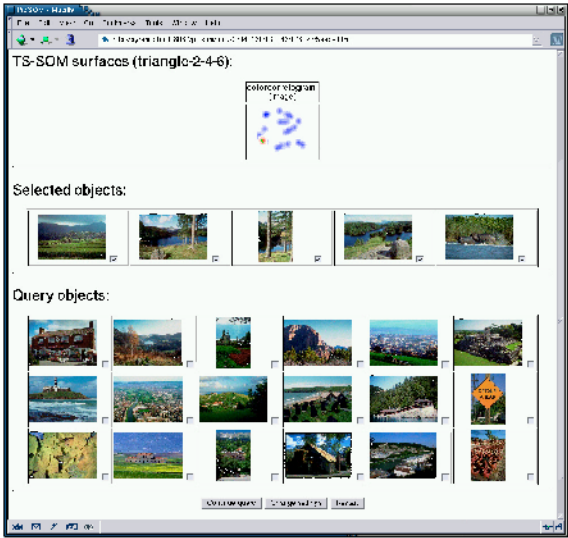
## 2.2 PicSOM

The PicSOM system is a framework for generic research on algorithms and methods for CBIR, using the self-organizing map as the basic image indexing method. A more detailed description of the system and results of experiments performed with it can be found in [7,8,9]. For computational reasons, PicSOM uses a special form of the SOM algorithm, the Tree Structured Self-Organizing Map (TS-SOM) [6]. The hierarchical structure of TS-SOM reduces the complexity of training large SOMs by exploiting the hierarchy in finding the best-matching map unit (BMU) for an input vector. In addition, the produced hierarchical representation of the image database can be utilized in browsing and visualizing the images in large databases.

The main image retrieval method of PicSOM is query by examples in which the image query is iteratively refined based on the user's relevance feedback. The relevance information is mapped from the images to the corresponding BMUs on the used SOMs and then spread to the map neighborhoods. This way, one obtains areas of positive and negative responses, which are illustrated with red and blue colors, respectively, on the user interface. For determining the images shown on the next query



round, PicSOM supports multiple features as the responses from the parallel SOMs are combined automatically, although only one feature was used in this study.



**Fig. 2.** The graphical user interface of the PicSOM system.

Fig. 2 shows the user interface of PicSOM during an example query for landscape images based on a color correlogram feature. First, the responses of the user's previous relevance evaluations are visualized on the feature map. The currently relevant-marked images are shown next and the images returned as best-scoring ones on this query round are shown below. The checkboxes beside the images are used for marking the relevance of the images to the current query. In addition, the user can at any time switch to image browsing by clicking on interesting locations on the used SOMs. Then, the corresponding portion of the SOM surface is displayed with a navigational aid for further browsing.

### 3 Task-Based User Evaluation

The performance and usability of the browsing systems was evaluated with a task-based user evaluation. The goal of the evaluation was to quantify how well the users are able to carry out various tasks solely based on automatic content-based methods, without prior manual annotation or categorization of the database. Further, the evaluation was expected to identify possible usability problems and provide valuable feedback for improvements.

### 3.1 Test Arrangement

**Setup.** The user evaluation involved 20 test users, mostly research personnel and a few students. The test users were required to have smooth computer skills and absolutely no prior experience of either IIRo or PicSOM.

The image database used in the evaluation contained 10144 images from 150 different subject matters in the CorelGALLERY collection. The subject matters were chosen so that the resulting database would be versatile and representative of a typical end user image database. Color correlograms extracted from the images were used as the metadata in IIRo and PicSOM. ACDSee does not utilize any content-based methods, but visualizes the images in a random layout.

The evaluation was conducted on a regular desktop PC, which had a 19 inch color monitor set to 1600x1200 pixel resolution. The graphical user interface of each system was set to cover the complete screen. Further, each system was adjusted to have the same background color, to eliminate any impact by contrast differences. The users had a mouse and a keyboard at their disposal.

The IIRo system used one multi-resolution SOM, where the root level had 90x90 nodes. The query-by-example functionality was disabled, to force the users to rely on the browsing method exclusively. The size of the browsing view was set to 100 images as a 10x10 grid. The size of the thumbnail images on the browsing view was set to 100x100 pixels.

The browsing view of the PicSOM system was set to 6x5 images, so that in terms of number of images it was roughly identical to the zoom view of the IIRo system (30 vs 33). This number of images could also be represented in the user interface simultaneously without scroll bars. The size of the thumbnail images was 120x90 pixels, the default setting of the PicSOM system.

In the ACDSee system the size of the thumbnail images was set to 100x100 pixels. Before the start of the test all images were loaded to the ACDSee so that the system did not have to load them during the browsing. The initial placement of the images on the browsing view was drawn randomly for each test user.

**Test procedure.** The 20 test users were first randomly divided into two groups (A and B) of 10 subjects. Group A evaluated first PicSOM and then IIRo, and group B vice versa, to eliminate the effect of learning. ACDSee was always the last system evaluated. At the beginning the test users were provided with written instructions and an introductory search task to familiarize them with the browsing system in question. Once the introductory task was completed, the actual evaluation commenced.

Each test user was asked to carry out five tasks described below. Each task was defined on a separate sheet of paper which the test user was allowed to study for an arbitrary amount of time. Once the test user told to be ready to carry out a task, the UI of the system being evaluated was exposed to the user. A task was deemed completed once the target image(s) had been found. The test user was also allowed to forfeit a task. After having completed the fifth and last task, the test user was asked to fill in a paper questionnaire. Having completed the questionnaire for one system the test user moved on to the evaluation of the next system. The ACDSee system was evaluated

only with task T1, since completing all five tasks would have taken too much time. After evaluating all three systems the test user filled in a second questionnaire, where (s)he was asked to rank the three systems, and explain the ranking.

**Tasks.** The test users were asked to carry out the following five tasks:

- T1: Find the image illustrated in Fig. 3,
- T2: Find an image of night-time sky,
- T3: Find five images of desert,
- T4: Find an image of the Statue of Liberty against a blue sky,
- T5: Find an image of a violin.



**Fig. 3.** Target image used in task T1.

## 3.2 Results

The following measurements were recorded:

- $t$  search time elapsed in carrying out a single task,
- $V$  the number of new images visualized in the UI during a single task,
- $T$  user's subjective assessment of the tryingness of the performance of a single task on scale 1=effortless ... 8=very trying,
- $F$  the number of forfeited tasks per user,
- $S$  user's subjective assessment of the satisfaction of the functionality of IIRo and PicSOM systems on scale 1=very unsatisfied ... 8=very satisfied.

In the following analysis symbols  $R_{group}^{system}$  refer to the results obtained with group ( $A$  or  $B$ ) for system ( $A$ =ACDSee,  $I$ =IIRo,  $P$ =PicSOM). We first study  $R_A^P$  and  $R_B^I$ , which are regarded as the most unbiased results, since they do not include any effect of learning by carrying out the tasks with the other system beforehand.

Fig. 4(a) shows the median task wise search times for PicSOM and IIRo and their average (we use median to eliminate the effect of 'outliers'). We see that in the first

four tasks IIRo provides 10-20 seconds shorter search times, whereas task T5 was very difficult for IIRo. The reason for this was the task definition, which intentionally did not include any reference to the desired color distribution. PicSOM's relevance feedback mechanism also proved useful in this task, resulting in a quicker convergence of the search space.

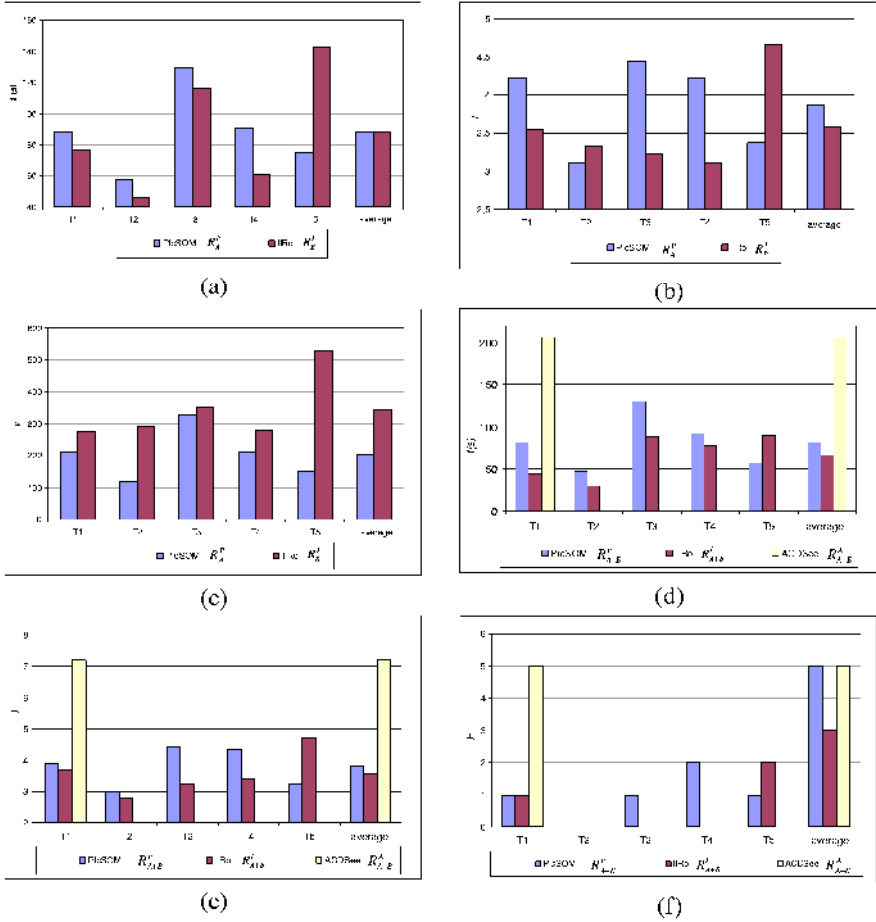


Fig. 4. Assorted results (see text for explanation).

Fig. 4(b) shows the task wise tryingness estimates and their average. They have roughly the same pattern as search times, excluding task T3, where given search times IIRo is judged to be surprisingly effortless relative to PicSOM. The reason is that IIRo typically shows simultaneously a number of similar images. Hence, having found one desired image the user is likely to have found several of them (just as was the setting in task T3), which compensates for the possibly long search time in finding the first correct image.

Fig. 4(c) shows the median of the new images shown to the user in the UI in each task and their average. We see that on average with IIRo the user is shown 140 images more than with PicSOM. This is mostly explained by the fact that the starting screen of IIRo has 70 images more than that of PicSOM.

Next we study results compiled over all 20 test users, i.e. results obtained for groups A and B are aggregated. Fig. 4(d) shows the median search times for each tasks and their average. We see that in task T1 the median search time for ACDSee was about three times longer than for IIRo or PicSOM, although the task was straightforward in the sense that spotting the desired image should have been easy due to its distinct colors. In task T4 the difference would probably have been even greater in favor of the content-based browsing systems.

The aggregate tryingness estimates in Fig. 4(e) show that IIRo and PicSOM are found roughly equally effortless to use, whereas ACDSee is found very trying. This is also demonstrated by the number of forfeited tasks shown in Fig. 4(f). Five out of 20 test users gave up in task T1, while IIRo and PicSOM scored only three and five forfeits in total, respectively. The poor results for ACDSee demonstrate the usefulness of content-based information in browsing a large image database.

The learning taking place during the evaluation can be seen by comparing the median search times of the two groups. The group evaluating a system as the second system scored lower median search times than the group evaluating that system as the first system, in case of both systems. If we look at the rankings of the three systems, in both groups the content-based system that was evaluated first got a slightly worse total ranking than the system evaluated second. Similarly, on average test users gave a slightly larger satisfaction estimate to the system evaluated second (PicSOM 5.0→5.6, IIRo 5.2→5.3). Possibly, these can also be attributed to learning. ACDSee system was ranked as the last one by all test users.

### 3.3 Guidelines for System Design

The user evaluation and the feedback from the test users produced important suggestions for system design. Too small thumbnails (100x100 pixels on 19 inch monitor set to 1600x1200 resolution) were the biggest shortcoming in both IIRo and PicSOM. Another major problem reported by the test users was the ‘semantic gap’ between the color-based clustering of images by the systems and the high-level categorization of images the users preferred to impose.

Following usability problems were identified for PicSOM: difficulties in finding a suitable example image among the images in the starting screen (could be addressed by using a larger starting screen), the user may forget to remove bad images from the collection of query images, which results in weaker performance, poor design of the button used for selecting an image as a good example, and no “return to beginning” button.

Following usability problems were identified for IIRo: zoom view occluded too many images in the browsing view, difficulties in recognizing that a node contained multiple images, mouse buttons were overloaded with too many functions, and no possibility to reverse to the previous visualization.

## 4 Conclusion

This paper presented a thorough task-based user evaluation of two content-based image database retrieval systems and the commercial ACDSee program. The results show that content-based information is very useful in browsing a large image database. Useful guidelines for future system design were also obtained.

## References

1. Baeza-Yates, R., Ribeiro-Neto N: Modern Information Retrieval. Addison Wesley, Essex (1999)
2. Card, S.K., Mackinlay, J.D., B. Shneiderman: Readings in Information Visualization, Using Vision to Think. Morgan Kaufmann Publishers, San Fransisco (1999)
3. Heesch, D., Yavlinsky, A., Rüger, S.: Performance Comparison Between Different Similarity Models for CBIR with Relevance Feedback. Proc. International Conference on Image and Video Retrieval, Urbana-Champaign (2003) 456-466
4. Hull, D.: Using Statistical Testing in The Evaluation of Retrieval Experiments. Proc. ACM SIGIR 1993, Pittsburg (1993) 329-338
5. Kohonen, T.: Self-Organizing Maps. Springer-Verlag, Heidelberg (2001)
6. Koikkalainen, P., Oja, E.: Self-organizing Hierarchical Feature Maps. International Joint Conference on Neural Networks, San Diego (1990) 279-284
7. Koskela, M.: Interactive Image Retrieval using Self-Organizing Maps. PhD thesis, Laboratory of Computer and Information Science, Helsinki University of Technology (2003), available online at: <http://lib.hut.fi/Diss/2003/isbn9512267659/>
8. Laaksonen, J., Koskela, M., Laakso, S., Oja, E.: Self-Organizing Maps as a Relevance Feedback Technique in Content-Based Image Retrieval. Pattern Analysis and Applications 4 (2001) 140-152
9. Laaksonen, J., Koskela, M., Oja, E.: PicSOM- Self-Organizing Image Retrieval with MPEG-7 Content Descriptions. IEEE Transactions on Neural Networks 13 (2002) 841-853
10. Nielsen, J.: Usability engineering. Academic Press, Boston (1993)
11. Matinmikko, E.: Image Database Browsing System. M.Sc. thesis, Department of Electrical Engineering, University of Oulu, Finland (2004)
12. Reeves, T., Hedberg, J.: Interactive Learning Systems Evaluation. Educational Technology Publications, Englewood Cliffs (2003)
13. van Rijsbergen, C.J.: Information Retrieval. Butterworths, London (1979)
14. ACDSee Photo Software 4.0, <http://www.acdsee.com> (2003)

# The CLEF Cross Language Image Retrieval Track (ImageCLEF) 2004

Paul Clough<sup>1</sup>, Mark Sanderson<sup>1</sup>, and Henning Müller<sup>2</sup>

<sup>1</sup>Department of Information Studies, University of Sheffield, Regent Court, 211 Portobello Street, Sheffield, S1 4DP, UK.

{p.d.clough,m.sanderson}@sheffield.ac.uk

<sup>2</sup>University Hospitals of Geneva, Division of Medical Informatics, 21 rue Micheli-du-Crest, CH-1211 Geneva 4, Switzerland.

henning.mueller@dim.hcuge.ch

**Abstract.** In this paper we describe ImageCLEF<sup>1</sup>, the cross language image retrieval track of the Cross Language Evaluation Forum (CLEF<sup>2</sup>). We instigated and ran a pilot experiment in 2003 where participants submitted entries for an ad hoc bilingual image retrieval task on a collection of historic photographs from St. Andrews University Library. This was designed to simulate the situation in which users would express their search request in natural language but require visual documents in return. For 2004 we have extended the tasks to include a medical image retrieval task and a user-centred evaluation.

## 1 Introduction

A great deal of research is currently underway in the field of Cross Language Information Retrieval (CLIR) where documents written in one language are retrieved by a query written in another (see, e.g. [11] and [16]). One can consider CLIR as basically a combination of machine translation (MT) and traditional monolingual information retrieval (IR). Most CLIR research has focused on locating and exploiting translation resources with which the user's search requests or target documents (or both) are translated into the same language. Campaigns such as the Cross Language Evaluation Forum (CLEF) [16] and the Text REtrieval Conference (TREC) [20] multilingual track have helped encourage and promote international research, as well as create standardised resources for CLIR evaluation.

However, one area of CLIR research which has received less attention is image retrieval. In collections such as historic or stock-photographic archives, medical case notes and art/history collections, images are accompanied by some kind of text (e.g. metadata or captions) semantically related to the image [2][12]. Images can then be retrieved using standard IR methods based on textual queries. However, retrieval from an image collection offers distinct characteristics from one in which the document to be retrieved is natural language text [1][10]. For example, the way in which a query is formulated, the method used for retrieval (e.g. based on low-level features derived

---

<sup>1</sup> ImageCLEF: <http://ir.shef.ac.uk/imageclef2004/>

<sup>2</sup> CLEF: <http://www.clef-campaign.org>

from an image, or associated text), the types of query, how relevance is assessed, the involvement of the user during the search process, and fundamental cognitive differences between the interpretation of visual versus textual media. Methods of image retrieval are typically based on visual content<sup>3</sup> (e.g. colour, shape, spatial layout and texture), or by text/metadata associated with the image (see, e.g. Smeulders et al. [18] and Goodrum [10]).

For those organisations managing image repositories in which text is associated with images (e.g. on-line art galleries), one way to exploit these is by enabling multi-lingual access to them. To promote research in this area we instigated ImageCLEF [5] as part of the CLEF campaign. We felt this contribution would address an important and timely problem not dealt with by existing cross language evaluation. We envisage ImageCLEF will appeal to both commercial and academic research communities including: cross language information retrieval, image retrieval, and user interaction. The main aims of the ImageCLEF campaign are: (1) to promote and initiate international research for CL image retrieval, (2) to further our understanding of the relationships between CL texts and images for IR, and (3) to create a set of useful standardised resources for CL image retrieval to scientific communities in the whole.

The paper divides into the following: in section 2 we describe the ImageCLEF 2003 test collection for an ad hoc retrieval task, in section 3 we describe tasks offered in ImageCLEF 2004 and finally in section 4 we summarise the contents of this paper and provide some ideas for future work in cross language image retrieval.

## 2 Building a Test Collection for Multilingual Image Retrieval

Evaluation of retrieval systems is either system-focused, e.g. comparative performance between systems or user-centered, e.g. a task-based user study. For many years IR evaluation has been dominated by comparative evaluation of systems in a competitive environment. The design of a standardised resource for IR evaluation was first proposed over 30 years ago by Cleverdon [4] and has since been used in major IR conferences such as TREC [20], CLEF [16] and NTCIR [3]. Over the years the creation of a standard test environment has proven invaluable for the design and evaluation of practical retrieval systems both within and outside a competitive environment. The main components of a TREC-style test collection are: (1) document collection, (2) topics, and (3) relevance assessments.

In TREC, NTCIR and CLEF, participants are given test collection data and topics and asked to submit their entries. A subset, chosen by the organisers, is used to create document pools, one for each topic. Domain experts (assessors) are then asked to judge which documents in the pool are relevant or not. Document pools are created because in large collections it is infeasible to judge every single document for relevance. These assessments are then used to assess the performance of submitted systems. User-centred evaluation is important to assess the overall success of a retrieval system which takes into account other factors other than just system performance, e.g. the design of the user interface and system speed (Dunlop argues this in [7]). A number of researchers have highlighted the advantages of user-centred evaluation, particularly in image retrieval systems (see, e.g. [10], [14] and [7]). One of the main aims of ImageCLEF is to provide both the CLIR and image retrieval communities a num-

---

<sup>3</sup> These are called Content-Based Information Retrieval (CBIR) systems.



ber of useful resources (datasets and relevance assessments) to facilitate and promote further research in multilingual image retrieval.

Calls for a TREC-style evaluation for image retrieval systems have been suggested [10][15][19], although Forsyth [9] argues that the evaluation of CBIR systems at the moment is useless because systems are too bad (hence the interest in combining both textual and visual features). We are unaware of existing test collections for CL image retrieval, although evaluation resources do exist to evaluate specific image retrieval tasks, e.g. journalism [13] and CBIR systems, e.g. Benchathon<sup>4</sup>. One of the largest obstacles in creating a test collection for public use is securing a suitable collection of images for which copyright permission is agreed. This has been a major factor influencing the datasets used in the ImageCLEF campaigns. The ImageCLEF test collection provides a unique contribution to publicly available test collections and complements existing evaluation resources.

## 2.1 The Existing ImageCLEF Test Collection

Because CL image retrieval encompasses at least two research areas: (1) image retrieval and (2) CLIR, building a comprehensive and suitable test collection is a tall order. Therefore, in 2003 we organised a pilot experiment at CLEF with the following aim: given a multilingual statement describing a user need, find as many relevant images as possible. More formally the task was a bilingual ad hoc retrieval task in which a static collection was searched using previously unseen topics.

The retrieval task was designed to simulate the situation in which a user expresses their need in a language different from the collection, requiring a visual document to fulfil their search request (e.g. searching an on-line art gallery or stock photographic collection). For this retrieval task query translation is the preferred method of bridging the language gap as translating the collection would be both time and resource expensive and less likely in practice. Participants were not constrained in their use of retrieval method, enabling either text or content-based searches (or a combination of both). As a retrieval task there are several challenges other than translation which include: (1) captions typically short in length, (2) images of varying content and quality, (3) bridging the gap between colloquial and domain-specific language used in the captions and cross language queries, and (4) queries short in length thereby providing little context for translation.

The dataset used consisted 28,133 historic photographs from the library at St Andrews University [17]. All images are accompanied by a caption consisting of 8 distinct fields which can be used individually or collectively to facilitate image retrieval (see Fig. 1). The 28,133 captions consist of 44,085 terms and 1,348,474 word occurrences; the maximum caption length is 316 words, but on average 48 words in length. All captions are written in British English and contain colloquial expressions and historical terms. Approximately 81% of captions contain text in all fields, the rest generally without the description field. In most cases the image description is a grammatical sentence of around 15 words. The majority of images (82%) are black and white, although colour images are also present.

---

<sup>4</sup> <http://www.benchathon.net/>

Record ID: JV-A.000460  
 Short title: The Fountain, Alexandria.  
 Long title: Alexandria. The Fountain.  
 Location: Dunbartonshire, Scotland  
 Description: Street junction with large ornate fountain  
 with columns, surrounded by rails and  
 lamp posts at corners; houses and shops.  
 Date: Registered 17 July 1934  
 Photographer: J Valentine & Co  
 Categories: [ columns unclassified ][ street lamps - or-  
 nate ][ electric street lighting ][ shepherds &  
 shepherdesses ][ streetscapes ][ shops ]  
 Notes: JV-A460 jf/mb



**Fig. 1.** An example image and caption (see: <http://www-library.st-andrews.ac.uk>).

We generated fifty representative search requests in English (called *topics*) and translated them into 6 different languages: Dutch, Spanish, German, French, Italian and Chinese (provided by the National Taiwan University or NTU). In TREC, CLEF and NTCIR final topics are chosen from a pool of suggestions generated by searchers familiar with the domain of the document collection. Frequently searched subject areas in the St Andrews were identified by analysing log files generated from accesses to a web search engine used by the library. Based on these subject areas we created queries that would test the capabilities of both a translation and image retrieval system, e.g. pictures of specific objects versus pictures containing actions, broad versus narrow concepts, topics containing proper names, compound words, abbreviations, morphological variants and idioms. Each topic consisted of a short title, a longer narrative describing the search request and an exemplar relevant image. For ImageCLEF 2003 only topic titles were translated due to limited resources available to us.

## 2.2 Relevance Assessments and Evaluation

What turns a set of documents and queries into a test collection are the relevance judgments, manual assessments of which documents are relevant or not for each topic. Judging whether an image is relevant or not is highly subjective (e.g. due to knowledge of the topics or domain, different interpretations of the same document, and searching experience), therefore to minimise this two assessors judged each topic.

We adopted the pooling method as used in TREC, CLEF and NTCIR where a set of candidate documents is created (called the *pool*) by merging together the results of the top  $n$  documents from the ranked lists provided by participants. This assumes that highly ranked documents from each entry will contain relevant documents. Ideally, ranked lists should come from a diverse range of systems to ensure maximal coverage. We also supplemented the pooling method with manual interactive searches (also known as *interactive search and judge* or ISJ) to ensure good quality pools (as used in NTCIR). We found assessors were able to judge the relevance of images very quickly (especially eliminating non-relevant ones) enabling *all* ImageCLEF submissions to be used in creation of the pools (compared to a subset of runs for text-based assessment). One of the authors familiar with the collection assessed all fifty topics to provide a „gold“ set of judgments; in addition, ten assessors from the University of Sheffield judged five topics each to provide a second judgment for each topic using a custom-built assessment tool.

Images were judged relevant if *any* part of the image was deemed relevant. Primary judgment was made on the image, but assessors also consulted the image captions. Assessors were asked to judge the relevance of images using a ternary scheme: relevant, partially relevant and not relevant to deal with potential uncertainty in the assessor's judgment (i.e. it is possible to determine that the image is relevant, but less certain whether it exactly fulfils the need described by the topic). Unlike other test collections we provided four sets of relevance assessments (called *qrels*) - strict/relaxed union/intersection - with which to assess system performance based on the overlap of relevant images between assessors and whether the relevance sets include images judged as partially relevant or not. These are further described in [5]. The strict relevance set can be contrasted with a high-precision task; the relaxed set providing an assessment that promotes higher recall.

## 2.3 Results and Lessons Learned

Four groups entered ImageCLEF 2003: Sheffield University, NTU, University of Surrey and Daedalus, a Spanish R&D organization. All participants used text-based retrieval methods with no content-based image analysis. Results from ImageCLEF have shown that in general CL image retrieval using query translation can achieve relatively high performance for the suggested bilingual search task. However, we found retrieval performance to vary dramatically across both language and topic. The highest result was obtained for French (78% of monolingual); the worst for Chinese (51% of monolingual) indicating there is still room for improvement. In particular, enhancement to deal with poor retrieval caused by translation errors is required. Results from ImageCLEF showed: for Chinese retrieval transliteration of proper names was beneficial, and for other languages thesaurus-based query expansion improved performance. ImageCLEF was effective at attracting new research groups to CLEF and this year is advertised as an entry-level CLIR task.

Based on our experiences from last year we have made the following changes to the ImageCLEF track: (1) to offer greater diversity we have added a medical retrieval task, (2) to promote ImageCLEF as an entry-level CLIR task we are offering topics in 12 languages rather than 6, (3) to encourage participants to exploit visual features we have setup public access to a default CBIR system, (4) due to ambiguity in relevance assessments we have selected more specific topics including queries refined by photographer, location and date (general queries such as „mountain scenery“ retrieved too many images and were too laborious to assess), and (5) we are using relevance assessors familiar with the collection (this includes native English speakers who are familiar with colloquial English/Scottish terms, e.g. „perambulator“).

## 3 The ImageCLEF 2004 Track

### 3.1 The Bilingual Ad Hoc Retrieval Task

A bilingual ad hoc task similar to that run in 2003 is being offered to participants to enable further experiments on the St. Andrews dataset and determine whether improvements can be made on last year's results. Experiments will compare: (1) differ-

ent methods of query translation (e.g. dictionary-lookup versus MT), (2) query expansion (e.g. global versus local methods), (3) the use of text-based and CBIR methods used either separately or combined, (4) different retrieval models, (5) different indexing methods (e.g. indexing all or some fields) and (6) manual vs. automatic relevance feedback.

A new set of 25 topics has been produced in the same manner as before (decide on general topics and then refine). However, in addition to using St. Andrews query logs, we also used subject areas supplied by staff from St. Andrews' library. Topic refinement is based on the query categorisation scheme suggested by Armitage et al. [1] for picture archives and designed to test a range of different CL and image search parameters. Topics have been translated into the previous languages, plus Japanese, Danish, Russian, Finnish, Swedish and Arabic. One non-intentional but interesting „feature“ of translated topics in ImageCLEF 2003 was the introduction of translation errors, e.g. spelling mistakes and erroneous diacritics, resulting in low retrieval performance for some topics. These problems are not addressed by existing CLEF tasks. We will provide two sets of topics: one set will contain spelling errors; the other will be checked and free of such errors.

### 3.2 The Medical Image Retrieval Task

To offer participants a different domain/scenario and encourage the use of CBIR system we have introduced a task based on medical retrieval. In the ad hoc task it is the query which is multilingual; in the medical retrieval task the document collection is multilingual presenting different CLIR challenges.



**Fig. 2.** Example images from the CasImage dataset (<http://www.casimage.com/>)

In general, medical practitioners are unsatisfied with retrieving images by text and the implicit knowledge stored in the images plus attached text is rarely used. As a diagnostic aid, being able to search a database of images with a new example would enable them to obtain more evidence. The goal of this task is to investigate the use of CBIR and text-based retrieval systems for this kind of medical retrieval task. The task is being run by University Hospitals of Geneva who are supplying the medical data, topics and relevance judgments. The medical task is this: given an example image, find similar images which will be helpful in confirming the initial diagnosis. Because the initial retrieval has to be visual, we expect the case notes to be useful in finding additional similar images complementary to CBIR. We also aim to evaluate whether

relevance feedback can improve performance, compare relevance feedback using either image/text or both, and whether images alone can be used for pseudo relevance feedback.

The dataset (CasImage) consists of 8,751 anonymised medical images, e.g. scans, and x-rays (see Fig. 2). The majority of images are associated with *case notes*, a written description of a previous diagnosis for an illness the image identifies. Case notes consist of several fields including: a diagnosis, a description, clinical presentation, keywords and title. The task is multilingual because case notes are mixed language written in either English or French. Not all case notes have entries for each field and the text itself reflects real clinical data in that it contains mixed-case text, spelling errors, erroneous French accents and un-grammatical sentences. In the dataset there are 2,078 cases to be exploited during retrieval (e.g. query expansion).

Currently 25 example images (topics) have been chosen as representative from the dataset. A set of ground truths for each topic has already been identified by domain experts based on the CBIR system developed by the third author<sup>5</sup> and these will form part of the document pools created from participant's entries. Pools will be formed in a manner similar to the ad hoc task and medical practitioners will help judge the relevance of the pools after final submissions. In this task images are judged using a binary relevant or not relevant judgement and assessments will be used to evaluate participant's entries. This retrieval task offers a number of challenges including: (1) combining text and content-based methods of retrieval after an initial visual search, (2) dealing with domain-specific medical terminology, (3) case notes of varying quality in more than one language (i.e. a mixed language index), and (4) the high cost of returning non-relevant images (i.e. mis-diagnosis) which is always inevitable when using visual-only search methods.

### 3.3 The Interactive Retrieval Task

Campaigns such as iCLEF<sup>6</sup> have shown the value of user-centred evaluation for CLIR and CL image retrieval would seem to be a rich source for user-centred experiments. Past research has shown that the search activities of a user in an image retrieval system vary between searching for specific images and browsing the image collection (see, e.g. [10] and [6]). For a CL image retrieval system, the issue is how best the system can support the user's search in locating relevant images as quickly, easily and accurately as possible. User-centered evaluation in a variety of contexts and domains will help us determine how CL image retrieval systems can best help users to: (1) formulate their queries (e.g. whether text or visual queries alone are best or can be used in combination), (2) refine the search request - query reformulation will depend on the outcome of the system and could involve refinements using textual and/or visual features, (3) browse the collection, and (4) identify relevant images (e.g. what additional information would help the user judge the relevance of an image and how best is this displayed).

Cox et al. [6] suggest three classes of image search: (1) target or known-item search (i.e. find a specific image), (2) category search (e.g. „find pictures of the Eiffel Tower“) and (3) open-ended browsing (i.e. wandering through the collection). They

<sup>5</sup> See <http://vipser.unige.ch/> for a list of publications about the VIPER CBIR system.

<sup>6</sup> See <http://terral.lsi.uned.es/iCLEF/> for information about iCLEF.

argue that the target search encompasses the other categories of search; it is simple for the user to perform and has clear measures of effectiveness. The goal for the user in such a task is given an image to find it again from the collection. Unlike being given a textual topic description, the user must interpret the given image and generate suitable query terms in a given language (different from the document collection). The scenario models the situation in which a user searches with a specific image in mind (perhaps they have seen it before) but without knowing key information thereby requiring them to describe the image instead, e.g. searches for a familiar painting whose title and painter are unknown. This task will use the St. Andrews dataset and our experimental setup will follow the guidelines for user-centred experiments as suggested by iCLEF. This task will be undertaken with collaboration from iCLEF organisers to ensure a consistency in CLEF methodologies. Participants are asked to follow the experimental setup but can perform whatever experiments they like.

A minimum of 8 users and 8 topics are required for this task. Users are given 10/15 minutes to find each image using only CL queries. Topics are general enough so that people unfamiliar with the collection can still perform the searches. Captions must also be translated into this language before being displayed (if at all) to the user. The aim of this experiment will be to observe users search habits and to determine what kind of interface best supports *query refinement*. For example the user is shown a picture of an arched bridge but starts with the query „bridge“. By finding similar images and maybe using keywords from their captions, the user refines the query until the relevant image is found. Query. Topics and systems will be presented to the user in combinations following a *Latin-square* design to ensure user/topic and system/topic interactions are minimised. Qualitative performance measures is captured using questionnaires provided by us, and quantitative measures include: whether the given image is found or not, the time taken to find the image, the number of images viewed before finding the image and number of user interactions required.

## 4 Conclusions and Future Work

In this paper we have discussed our proposal for three cross-language image retrieval tasks as part of the ImageCLEF campaign. The tasks vary across domain, scenario, where CLIR is used, whether content-based image retrieval is required and whether the task is system or user-centered. Results from ImageCLEF 2003 have shown CL image retrieval to be a success, but large improvements can still be obtained for some languages (e.g. Chinese). Our aim is to promote CL image retrieval and provide a standardised set of resources in the form of test collections (i.e. a collection, topics and relevance assessments) which can be used in further CL image retrieval experiments. In future work we plan to expand the collections and tasks offered in ImageCLEF. In particular we would like to offer collections with non-English captions provide a Web-based image retrieval task and offer further image retrieval tasks, e.g. aspectual retrieval.

## References

1. Armitage, L.H. and Enser, P.: Analysis of User Need in Image Archives. In *Journal of Information Science* **Vol. 23(4)** (1997) 287-299
2. Chen, F., Gargi, U., Niles, L. and Schütze, H.: Multi-Modal Browsing of Images in Web Documents. In *Proceedings of SPIE Doc. Recognition and Retrieval VI* (1999) 122-133
3. Chen, K., Chen, H., Kando, N., Kuriyama, K., Lee, S. and Myaeng, S.: Overview of CLIR Task, Third NTCIR Workshop, Japan (2002)
4. Cleverdon, C.W.: The Cranfield Tests on Index Language Devices. In: K. Spark-Jones and P. Willett (eds), *Readings in Information Retrieval*, Morgan Kaufmann (1997) 47-59
5. Clough, P. and Sanderson, M.: The CLEF 2003 Cross Language Image Retrieval Track. In *Proceedings of the Cross Language Evaluation Forum (CLEF) Workshop*, Norway (2003)
6. Cox, I.J., Miller, M.L., Omohundro, M. and Yianilos, P.N.: Target Testing and the PicHunter Bayesian Multimedia Retrieval System. In *Proceedings of Advanced Digital Libraries (ADL'96) Forum*, Washington D.C. (1996)
7. Dunlop, M.: Reflections on MIRA: Interactive Evaluation in Information Retrieval. In *Journal of the American Society for Information Science* Vol. 51(14) (2000) 126-1274
8. Flank, S.: Cross language Multimedia Information Retrieval. In *Proceedings of Applied Natural Language Processing and the North American Chapter of the Association for Computational Linguistics* (2000)
9. Forsyth, D.A.: Benchmarks for Storage and Retrieval in Multimedia Databases. In *Proceedings of SPIE International Society for Optical Engineering* **Vol. 4676** (2001) 240-247
10. Goodrum, A.A.: Image Information Retrieval: An Overview of Current Research. In *Informing Science* **Vol. 3(2)** (2000) 63-66
11. Grefenstette, G.: *Cross language Information Retrieval*. Kluwer Academic Publishers, Norwell, MA, USA. (1998).
12. Harmandas, V., Sanderson, M. and Dunlop, M.D.: Image Retrieval by Hypertext Links. In *Proceedings of the 20<sup>th</sup> ACM SIGIR conference* (1997) 296-303
13. Markkula, M. and Sormunen, E.: Searching for Photos – Journalist's Practices in Pictorial IR. In *Proceedings of Conference on Image Retrieval (CIR'98)* (1998)
14. McDonald, S., Lai, T.S., Tait, J.: Evaluating a Content Based Image Retrieval System. In *Proceedings of SIGIR'01* (2001) 232-240
15. Müller, H., Müller, W., McG. Squire, D., Marchand-Maillet, S. and Pun, T.: Performance Evaluation in Content-Based Image Retrieval: Overview and Proposals. In *Pattern Recognition Letters* **Vol. 22(5)** (2001) 593-601
16. Peters, C. and Braschler, M.: Cross Language System Evaluation: The CLEF Campaigns. In *Journal of the American Soc. for Inf. Sci. and Tech.* **Vol. 52(12)** (2001) 1067-1072
17. Reid, N.: The Photographic Collections in St Andrews University Library. In *Scottish Archives* **Vol. 5** (1999) 83-90
18. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A. and Jain, R.: Content-Based Image Retrieval at the End of the Early Years. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* **Vol. 22(12)** (2000) 1349-1380
19. Smith, J.R.: Image Retrieval Evaluation. In *Proceedings of the IEEE Workshop of Content-Based Access to Image and Video Databases* (2001) 112-113
20. Voorhees, E.M. and Harman, D.: Overview of TREC 2001, In *NIST Special Publication 500-250: Proceedings of TREC2001*, NIST. (2001)

# An Empirical Investigation of the Scalability of a Multiple Viewpoint CBIR System

James C. French\*, Xiangyu Jin, and W.N. Martin

Department of Computer Science, University of Virginia, Charlottesville, VA, USA,  
{french,xj3a,wnm}@cs.virginia.edu,  
<http://www.cs.virginia.edu/~cyberia>

**Abstract.** Our work in content-based image retrieval (CBIR) relies on content-analysis of multiple representations of an image which we term multiple viewpoints or channels. The conceptual idea is to place each image in multiple feature spaces and then perform retrieval by querying each of these spaces and merging the several responses. We have shown that a simple realization of this strategy can be used to boost the retrieval effectiveness of conventional CBIR. In this work we evaluate our framework in a larger, more demanding test environment and find that while absolute retrieval effectiveness is reduced, substantial relative improvement can be consistently attained.

## 1 Introduction

Content-based image retrieval (CBIR) has been the object of considerable study since the early 90's. Much effort has gone into characterizing the "content" of an image by means of a variety of features for the purpose of indexing and subsequent retrieval. In earlier work [1] we proposed a strategy to capitalize on this work and to extend it by employing content-analysis of multiple representations of an image which we term multiple viewpoints[2]. The idea is to place each image in multiple feature spaces and then effect retrieval by querying each of these spaces and merging the several responses. The impetus for this research comes from work in text IR on combination of evidence strategies that dates back to the early 90's. Two approaches have generally been used. In the first approach a diversity of queries is used to capture an information need more precisely. The several queries can be combined before searching, or issued individually and the results of each query merged afterwards. The work of Belkin et al.[3,4] adopts this approach.

The second strategy is to use a diversity of representations, that is, create several indexes over the same corpus of documents. The typical strategy is to index the corpus with the same technology varying indexing parameters, or to

---

\* This material is based upon work done while serving at the National Science Foundation. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.



index the corpus with different technologies. Queries are processed in each setting with the results being merged afterwards. The work of Fox and Shaw[5] adopts this strategy. Bartell et al. [6] also look at combining evidence in this framework. The approach we adopt for extending CBIR systems to combine multiple evidence is analogous to this latter approach.

These ideas are embodied in our *synthetic retrieval model* for CBIR[7] shown schematically in Figure 1. We refer to this as a synthetic retrieval model because we merge the various viewpoints and synthesize a channel for presentation to the user. We increase the number of viewpoints in CBIR systems in three different ways: multiple representations; multiple CBIR systems; and multiple queries. We also employ relevance feedback to further increase retrieval performance. Within this framework we have investigated the use of a diversity of representations that we call *channels* to achieve retrieval effectiveness gains over conventional CBIR[1,8,9,7]. Our approach is exogenous; we treat the CBIR system as a black box. We create additional channels by transforming the images and indexing the transformed images. Our four channels derive from the color positive (C+) and negative (C-) and the black and white positive (B+) and negative images (B-). Note that the C+ channel corresponds to the conventional CBIR system.

In this paper we evaluate our framework in a larger, more demanding test environment. Due to page limitations we refer the reader to [1,8,9,7] for specific details of our framework. In the remainder of this paper we describe the current experimental setup and finally discuss our results.

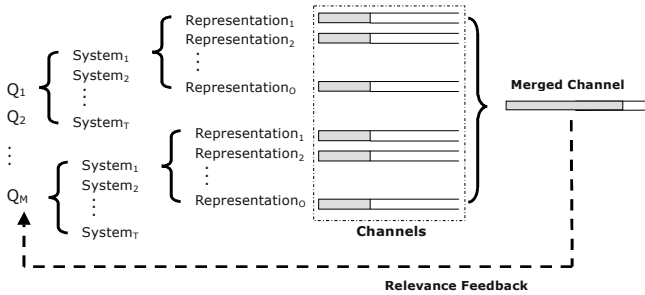


Fig. 1. Synthetic Retrieval Model

## 2 Experimental Setup

### 2.1 Basic CBIR Technology

We used a basic CBIR setup similar to that used in the MiAlbum system used in the work of Liu et al. [10]. Our system uses seven image features: three color features and four texture features. For similarity comparisons each feature was compared separately and then combined with equal weight.

## 2.2 Testbed

**Test Data.** We used two different image collections in this work.

1. **D34**: this is 3,400 images drawn from 34 categories of the COREL image collection. Each category contains 100 images. The categories were chosen because each of the images has a salient foreground object.
2. **D594**: this is a larger version of the COREL database consisting of 594 image categories each having 100 images each. Thus, **D594** contains 59,400 images. It should be noted that **D34** is a proper subset of **D594**.

**Query Sets.** We use three different query sets in this work.

1. **Q3400**: Each of the images in **D34** is used as a query. Thus **Q3400** = **D34** and there are 3,400 query images.
2. **Q204**: The 34 categories of **D34** are uniformly sampled and 6 images are included in **Q204** from each category. This is a 6% sample of **D34** with equal representation of each category. Thus, there are 204 query images in **Q204**.
3. **Q3564**: The 594 categories of **D594** are sampled in the same way as **Q204**. Thus, this is a 6% sample of **D594** with 3,564 queries and equal representation of all categories in the sample. Note that **Q204** is not a proper subset of **Q3564** and has no specific relation to it.

**Ground Truth.** In earlier work we used the “foreground” groundtruth for **D34**[1,8,9], but since we do not have the equivalent for the additional image categories in **D594**, we have used a different but consistent “COREL” groundtruth in the work reported here. This latter groundtruth is defined to mean that all the images in an image category are relevant to all the other images in the category and not relevant to any other images. Thus, any image selected from a test collection to act as a query will have exactly 99 relevant images in the collection.

Our earlier work has shown remarkable consistency between the performance as measured by these two groundtruths and we have never had one contradict the other in an experiment so we believe this choice to be adequate for our purposes.

**Indexing the Images.** We created four indexes corresponding to each channel in our testbed. The images were transformed into the representation of the channel and then indexed by our CBIR system. Thus, for each testbed we have a single corpus of images over which we have four separate indexes.

**Experiment Notation.** We denote a particular experiment by **Q/D** where **Q** is the query set and **D** is the testbed data set. For example, **Q204/D594** denotes the 204 queries of the 6% sample of **D34** processed against the 59,400 images in the large data set.

**User Model for Relevance Feedback.** In an earlier study [7] we observed a significant improvement in retrieval performance when using relevance feedback, that is, providing images identified as relevant in one iteration of the search to the query set in the next iteration. This is consistent with the results of other studies of relevance feedback. Our approach is to issue each feedback query

independently and then merge the results for presentation to the user. This leaves open the issue of how the feedback queries are chosen. We use two strategies:

1. Identify the top  $k$  images (Top- $k$ ); and
2. Take  $k$  images at random from among the relevant images (Random- $k$ ).

The former strategy is customarily used in text IR experiments. However, the latter strategy seems more appropriate for CBIR given the relative ease with which a user may judge the relevance of images. We feel that the Random- $k$  user model more accurately reflects user behavior. Earlier work [7] has shown that this strategy will result in higher retrieval performance because it defeats self-similarity in feedback images and therefore achieves a greater visual diversity among the feedback images. Note that there are at most  $k$  images chosen by either strategy because in some cases fewer than  $k$  images are present in the retrieval result. Further,  $k = 8$  in all the experiments reported here.

### 2.3 Methodology

The query processing is the same in all experiments. One query set is processed against one testbed. Each query is processed separately and the precision<sup>1</sup> at 100 images seen (P100) is calculated for each. The average P100 is calculated over the entire query set and that result is reported. We note that P20 has become a very common metric for reporting results in text-based IR. A typical CBIR UI displays 30-50 thumbnails at a time in response to a query. Because of the ease of evaluating images for relevance relative to text documents, we feel that P100 is a more appropriate performance measure. One hundred images is also the first time we could conceivably achieve recall<sup>2</sup> of 1.0 for any of the queries in our query sets.

Our merging results in [1,8,9] were produced using the *combSUM*[5,11] approach, that is, we summed the similarity values for images across the channels in which the image was included in the response set. (The conditions set out by Vogt[12] for linearly combining relevance scores apply here: our channels do have reasonable performance and they do not rank relevant documents similarly.) We have also used a rank sum approach, midrank merge<sup>3</sup>, for merging and found that to perform comparably with *combSUM*. We use that technique here.

## 3 Results

The four plots show in Figure 2(a-d) each show five experiments, **Q204/D34**, **Q3400/D34**, **Q204/D594**, **Q3400/D594** and **Q3564/D594** respectively. The

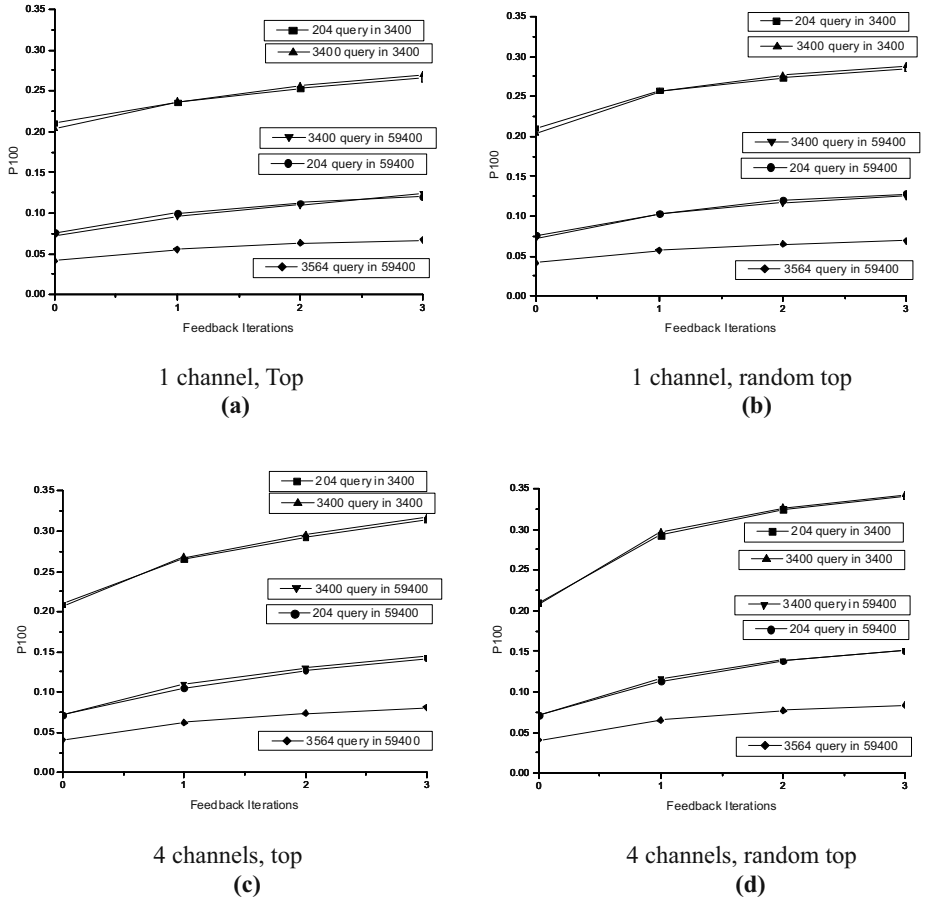
<sup>1</sup> Precision is the ratio of the number of relevant images retrieved to the total number of images retrieved.

<sup>2</sup> Recall is the ratio of the number of relevant images retrieved to the total number of relevant images.

<sup>3</sup> Each channel assigns a rank to each image retrieved and not retrieved. The assigned ranks are summed to determine the image's rank in the final result. When a channel does not retrieve an image, it is assigned a rank higher than 100.

plots are paired vertically by user model (Top- $k$ , Random- $k$ ) and are paired horizontally by channel configuration (one channel, four channels). The Random- $k$  user model is higher performing than the Top- $k$  model. We have observed this in earlier experiments [7] and attribute it to greater visual diversity in the feedback images. The topmost pair of lines show that the small sampled query set (**Q204**) has very similar performance to the larger query set (**Q3400**) in the smaller testbed (**D34**). The next two lines show that the small sampled query set also has very similar performance to the larger query set in the larger testbed (**D594**). This is consistent in all four plots.

*Conclusion 1: The smaller sampled query set, **Q204** is representative of the larger query set, **Q3400**, as regards performance evaluation.*



**Fig. 2.** Retrieval performance as measured by sample vs. all queries in small and large testbeds.

**Table 1.** Retrieval precision and performance increase after each feedback iteration (Top- $k$  user model). Avg. precision small (large) testbed is 29.3% (61.8%).

	<b>Q204</b>		<b>Q3400</b>		<b>Q3564</b>	
<b>Iteration</b>	<b>D34</b>	<b>D594</b>	<b>D34</b>	<b>D594</b>	<b>D594</b>	
0	.2109	.0759	.2032	.0730	.0413	
1	.2362 12.0%	.1003 32.1%	.2372 16.7%	.0966 32.3%	.0554 34.1%	
2	.2528 7.0%	.1126 12.3%	.2563 8.1%	.1102 14.1%	.0632 14.1%	
3	.2652 4.9%	.1209 7.4%	.2698 5.3%	.1190 8.0%	.0673 6.5%	
<b>Total</b>	25.7%	59.3%	32.8%	63.0%	63.0%	

The 4-channel configurations (Figure 2c,d) are equivalent to the single channel configurations (Figure 2a,b) initially but outperform them considerably in all feedback iterations.

*Conclusion 2: Relevance feedback in the multichannel configuration is more effective than in the single channel configuration.*

The four plots of Figure 2 clearly show that absolute retrieval effectiveness (as measured by P100) is lower in the larger database (**D594**) as compared with the effectiveness observed in the smaller (**D34**). This occurs in both single and multichannel configurations.

*Conclusion 3: CBIR retrieval precision is substantially reduced when the size of the database is increased.*

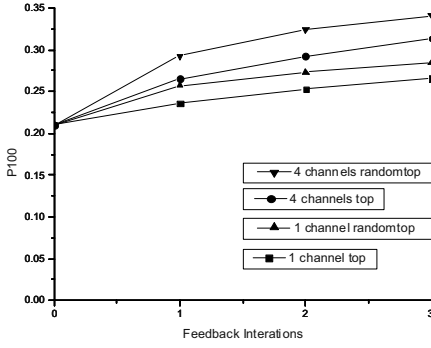
However, even though the absolute effectiveness is reduced in the larger testbed, the rate of improvement with each feedback iteration is roughly constant. In addition the overall improvement in each configuration was also very stable, averaging 62%. Table 1 shows the actual values. This is perhaps the most important feature of the multichannel approach.

*Conclusion 4: The multiple viewpoint techniques demonstrated in the smaller testbed (**D34**) are also effective in the larger testbed (**D594**).*

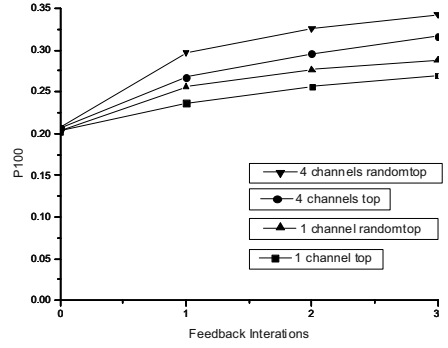
Finally, the **Q3564/D594** experiment has substantially lower performance than the **Q3400/D594**. Recall that all the queries in **Q3400** come from 34 of the 594 categories in **D594** whereas there are 6 queries from each of the categories of **D594** in **Q3564**. We hypothesize that **Q3400** is therefore an “easier” query set than **Q3564**.

*Conclusion 5: Q3400 and Q3564 do not have similar retrieval performance in D594.*

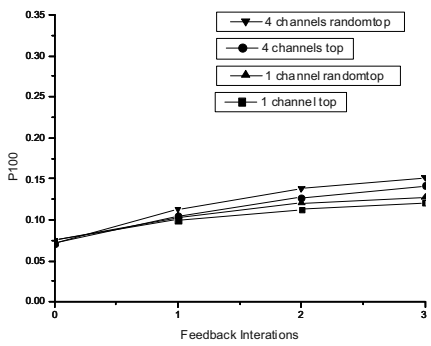
The four plots in Figure 3 are grouped vertically by query set with the smaller query set (**Q204**) on the left and the larger (**Q3400**) on the right. They are grouped horizontally by testbed size with the smaller testbed (**D34**) topmost and the larger (**D594**) on the bottom. In each case four lines are shown corresponding to the two user models (Top- $k$  and Random- $k$ ) and the two channel configurations (one, four). Again, the data support *Conclusion 1*. We are also led to the following conclusions.



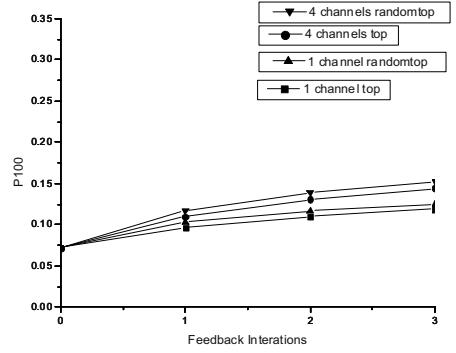
query 204 in 3400  
(a)



query 3400 in 3400  
(b)



query 204 in 59400  
(c)



query 3400 in 59400  
(d)

**Fig. 3.** Performance of single channel vs. multichannel CBIR for two user models.

*Conclusion 6: The Random- $k$  user model consistently achieves the highest precision.*

This is good news especially since we consider the Random- $k$  user model to more accurately reflect user feedback selection in real CBIR environments.

*Conclusion 7: The multichannel configuration is superior to the conventional single channel approach achieving greater precision with each feedback iteration.*

Especially noteworthy is the fact that this improvement is attainable in the larger image testbed, 59% (63%) for the smaller (larger) query set. We also observed this in the more difficult large test environment (**Q3564/D594**) discussed earlier where the query set is a 6% sample of the 59,400 images queried (see Table 1). The improvement (63%) was exactly the same as that achieved with the **Q3400** query set (see Figure 3c,d).

The results clearly indicate that even in the more difficult testbed, we can, in fact, combine the channels, even naively, to realize retrieval effectiveness gains over the conventional single-channel CBIR approach.

## 4 Conclusions

We have described a simple approach for improving the retrieval effectiveness of conventional CBIR systems. Our approach treats the CBIR technology as a black box which can be used to provide different channels of retrieval results for subsequent merging or for use in interactive retrieval interfaces. The channels are implemented as additional indexes over simple image transforms. This offers a simple, cost-effective strategy for boosting the performance of CBIR systems.

In [1] we showed that multichannel retrieval could increase CBIR retrieval effectiveness. We demonstrated an 8% increase in non-interpolated average precision with a 4-channel configuration of our CBIR system over the baseline system when ranking all the images of our test database. The average non-interpolated precision increases by 22% in the 4-channel system when we consider result lists of the top 100 images.

In [8] we looked at the potential for performance improvement when two CBIR systems were used to supply the viewpoints for constructing the synthetic channel. Again, the combination of multiple channels (this time from different CBIR systems) resulted in increased retrieval effectiveness. Moreover the combination of the two techniques, multiple systems and multiple representations, were complimentary and resulted in an even greater performance boost.

In [7] we looked at multiple queries as a means of achieving greater retrieval performance by providing more exemplars of the user's information need. We introduced the concept of visual diversity and examined the role of multiple representations and multiple CBIR systems in achieving visual diversity to improve retrieval with multiple queries. We also examined several strategies for accommodating relevance feedback in our synthetic framework. A new feedback evaluation strategy was also proposed and shown to be more effective because it increases the visual diversity of the feedback images.

In this paper we extended our work to validate our approach in a larger, more demanding retrieval environment. This kind of study is hampered somewhat by the lack of suitable testbeds with associated groundtruth. Our work here is extended to a testbed of 59,400 images from our earlier testbed of 3,400 images.

This study confirmed our earlier finding that the multiple viewpoint techniques, singly and in combination, improve retrieval effectiveness of CBIR systems even in more demanding retrieval environments. We found that although the absolute precision was reduced, the rate of improvement held up well in feedback iterations and the overall relative improvement after three feedback iterations was approximately 62%.

Another finding is that we can get extremely accurate performance evaluation with smaller query samples. This will enable us to conduct empirical studies more efficiently with confidence in the results.

Note that the techniques proposed here do not increase the user work in relevance feedback. The user is only concerned with the synthetic channel presented after each feedback cycle. The system transparently feeds the selected images back to all the underlying channels and merges the several results back into a synthetic channel for the user.

The synthetic retrieval framework also makes parallelization a simple matter. Retrieval on each channel is independent of all others. Thus, each channel can be assigned to a separate processor for query processing followed by a merging stage. This strategy can provide high retrieval efficiency in addition to improved retrieval effectiveness.

## References

1. French, J.C., Watson, J.V.S., Jin, X., Martin, W.N.: Using multiple image representations to improve the quality of content-based image retrieval. Technical Report CS-2003-10, Dept. of Computer Science, Univ. of Virginia (2003)
2. French, J.C., Chapin, A.C., Martin, W.N.: Multiple viewpoints: A strategy for searching multimedia content. In: Workshop on Multimedia Content in Digital Libraries. (2003)
3. Belkin, N., Cool, C., Croft, W., Callan, J.: The effect of multiple query representations on information retrieval system performance. In: Proc. of ACM SIGIR'93. (1993) 339–346
4. Belkin, N., Kantor, P., Cool, C., Quatrain, R.: Combining evidence for information retrieval. In: Proc. of TREC-2. (1994) 35–44
5. Fox, E., Shaw, J.: Combination of multiple searches. In: Proc. of TREC-2. (1994) 243–252
6. Bartell, B., Cottrell, G., Belew, R.: Automatic combination of multiple ranked retrieval systems. In: Proc. of ACM SIGIR'94. (1994) 173–181
7. Jin, X., French, J.C.: Improving image retrieval effectiveness via multiple queries. In: First ACM Inter. Workshop on Multimedia Databases. (2003) 86–93
8. French, J.C., Watson, J.V.S., Jin, X., Martin, W.N.: Integrating multiple multi-channel cbir systems. In: Proc. Inter. Workshop on Multimedia Information Systems (MIS 2003). (2003) 85–95
9. French, J.C., Watson, J.V.S., Jin, X., Martin, W.N.: An exogenous approach for adding multiple image representations to content-based image retrieval systems. In: Proc. Seventh Inter. Symp. on Signal Processing and its Applications. (2003)
10. Wenyan, L., Dumais, S., Sun, Y., Zhang, H., Czerwinski, M., Field, B.: Semi-automatic image annotation. In: Proc. of Human-Computer Interaction-Interact. (2001) 326–333
11. Shaw, J., Fox, E.: Combination of multiple searches. In: Proc. of TREC-3. (1995) 105–108
12. Vogt, C.: When does it make sense to linearly combine relevance scores. In: Proc. of ACM SIGIR'97. (1997)



# Real-Time Video Indexing System for Live Digital Broadcast TV Programs

Ja-Cheon Yoon<sup>1</sup>, Hyeokman Kim<sup>2</sup>, Seong Soo Chun<sup>1</sup>, Jung-Rim Kim<sup>1</sup>,  
and Sanghoon Sull<sup>1\*</sup>

<sup>1</sup>Department of Electronics and Computer Engineering, Korea University, Seoul, Korea  
{jcyoon, sschun, jrkim, sull}@mpeg.korea.ac.kr

<sup>2</sup>Department of Computer Science, Kookmin University, Seoul, Korea  
hmkim@kookmin.ac.kr

**Abstract.** In this paper, we introduce a real-time metadata service system that is implemented for live digital broadcast TV programs. The system is composed of three parts: an indexing host which indexes broadcast programs in real-time, a broadcaster where the segmentation metadata delivered from the indexing host is multiplexed into the broadcast stream and transferred to clients, and a client PVR that receives the metadata and locates a segment of interest from the recorded stream according to the time description of the delivered metadata. We propose to utilize broadcasting time for a time description of the segmentation metadata, so as to be free from the media localization problems in broadcast environment. In addition, we utilize a spatiotemporal visual pattern of a video for a verification tool of real-time indexing, such that we can reduce the false alarms of video segmentation caused by lack of an efficient tool for verifying video segment. As a result, we show the real experiments that are performed without requiring a return channel and demonstrate the feasibility of the proposed system.

## 1 Introduction

Recently, digital set-top boxes (STBs) with local storage known as a personal video recorder (PVR) begin to penetrate TV households. With this new consumer device, television viewers can record broadcast programs into the local storage of their PVR for viewing later. Due to the nature of digitally recorded video, viewers now have the capability of directly accessing to a certain point of recorded programs in addition to the traditional controls such as fast forward and rewind. Furthermore, if a segmentation metadata for a recorded program is available, the viewers can browse the program by selecting some of predefined video segments within the recorded program and play highlights as well as summary of the recorded program.

The metadata can be described in proprietary formats or in international open standard specifications such as MPEG-7 [1] or TV-Anytime [2]. The media location used in typical metadata such as TV-Anytime format are usually described by using either byte offset specifying the number of bytes to be skipped from the beginning of

---

\* Corresponding author

the file or media time specifying a relative time point from the beginning of the file. However, it might be ambiguous to describe a specific position of a broadcast stream using media time or byte offset, since it is hard to clearly identify when or where a program starts within the broadcast stream in which a number of programs or commercials are multiplexed and that is continuously being streamed without a program boundary marker through the broadcast network.

One possibility for random access to a specific position of broadcast streams is to use MPEG-2 DSM-CC Normal Play Time (NPT) [3] that provides a known time reference to a piece of media. For applications of TV-Anytime metadata in DVB-MHP broadcast environment, it was proposed that the NPT should be used for the purpose of time description [4, 5]. In the proposed implementation, however, it is required that both indexing system and client PVRs can handle NPT properly, thus resulting in highly complex controls on time.

Another possibility is to use the MPEG-2 Presentation Time Stamp (PTS) which indicates the time that a presentation unit is presented in the system target decoder. However, it requires parsing of packetized elementary stream (PES) layers, and thus it is computationally more expensive. Further, if a broadcast stream is scrambled, the descrambling process is needed to access to the PTS. Moreover, most of digital broadcast streams are scrambled, thus an indexing system cannot access the stream without an authorized descrambler if the stream is scrambled.

From a practical point of view, we propose to use broadcasting time as reference time, which is the simplest and most cost effective way of describing time index within a broadcast stream comparing to the above methods that require the complexity of implementation of DSM-CC NPT in DVB-MHP and computational cost and descrambling problems of PTS. Broadcasting time is carried on the broadcast stream in the form of system time table (STT) of ATSC [6] or time date table (TDT) of DVB [7]. Using broadcasting time as reference time does not require for an indexing system and client PVRs to be connected for synchronization through an interactive communication channel such as Internet. Also, it provides an efficient method to locate same position of the broadcast stream in both side of indexing system and client PVRs since the STTs or TDTs are contained in its temporal position of the broadcast stream according to the broadcasting time. For example, STT of ATSC is repeatedly broadcast once every second.

Fig. 1 shows the overall structure of proposed system composed of an indexing host (real-time indexing system: RTIS), a broadcaster, and a client PVR. A segmentation metadata for a live broadcast program is generated at the indexing host and delivered to the client PVR through the broadcasting network. The detailed descriptions will be shown in the following sections: the section 2 shows the detailed description of methods used in RTIS for the media localization and real-time segmentation, the section 3 presents the implementation of the test-bed and the experimental results, and the section 4 concludes the paper.

## 2 Media Localization and Real-Time Segmentation

We encounter two problems in implementing the proposed real-time metadata service scheme. One is how to localize the broadcast stream with broadcasting time in both

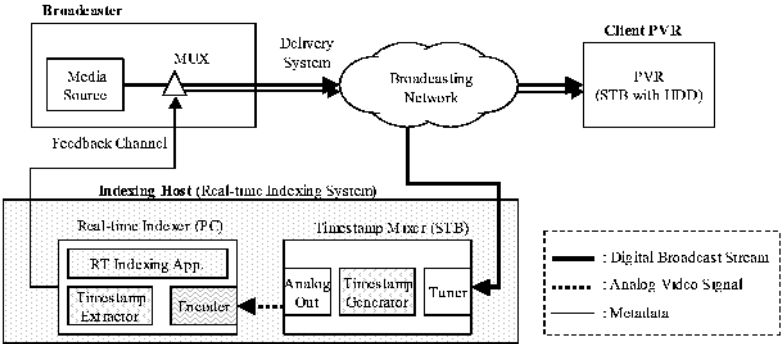


Fig. 1. Overall structure of test-bed for the real-time metadata service scheme

sides of the indexing system and the client PVRs. Another is how to index a live broadcast program in real-time, that is, how to detect shot boundaries (or scene changes) and group the shots into the segments of interest and how to easily verify the detected shot boundaries in real-time. The other problem is how to deliver the segmentation metadata to user's PVR in broadcast stream.

2.1 Media Localization Using Broadcasting Time

To solve the media localization problem in broadcast environments, we use the broadcasting time carried on STT or TDT of the broadcast stream in both sides of the indexing system and the client PVRs due to the convenient features of it as described in above section.

In the indexing system RTIS of Fig. 1, the timestamp mixer is introduced to index a digital broadcast stream with broadcasting time regardless of whether the stream is scrambled or not. The timestamp mixer superimposes the visual timestamp, such as a structured color-code [8], showing the current broadcasting time onto each frame of broadcast stream received through the tuner. The visually time-stamped analog output signals of the timestamp mixer are then encoded in low bit-rate at the real-time indexer. Using the stream encoded in low bit-rate, we can avoid a possible problem of directly accessing scrambled broadcast stream as well as a burden of indexing very high bit-rate stream such as HDTV broadcast stream.

In order to superimpose the timestamp for the current broadcasting time, the timestamp mixer examines broadcasting time carried on the STT or TDT of the received broadcast stream via its tuner.

In case of ATSC, it is recommended that I-frames shall be sent at least once every 0.5 second in order to have acceptable channel-change performance. Further, there exists a delay between the arrival time of a frame and its presentation time due to the VBV delay with maximum delay time of 0.5 second and decoding time delay. Fig. 2 shows an example of indexing and accessing the start position of a segment specified by the broadcasting time *BT* based on the above properties of ATSC.

The broadcasting time *BT* carried on the STT or TDT is represented with a discrete second unit. Thus the frames presented on screen during a discrete second have the

same broadcasting time with which they are time-stamped with the same broadcasting time

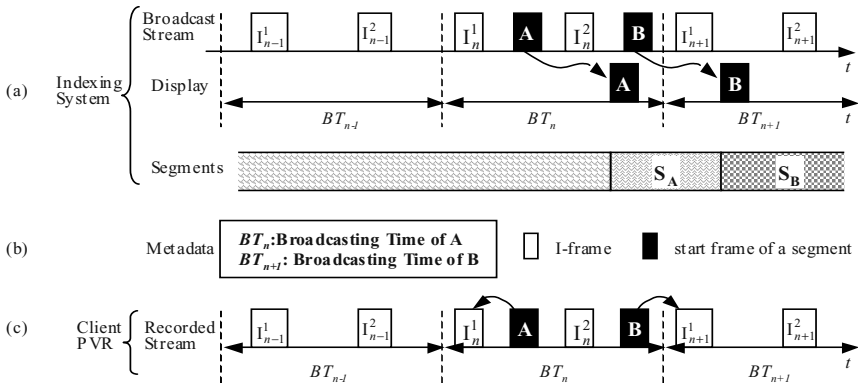
When the real-time indexer indexes the re-encoded video resulting from timestamp mixer, it extracts the broadcasting time for each video frame from the timestamp superimposed onto the frame. The extracted broadcasting time represents the current broadcasting time of the frame, at which the frame is presented on screen. For example, in case of frame A in Fig. 2(a), the broadcasting time of frame A is  $BT_n$  when the frame is displayed on screen on which the broadcasting time  $BT_n$  is time-stamped. Whereas in case of frame B, the broadcasting time of frame B is  $BT_{n+1}$  although the frame is arrived at previous time of  $BT_n$ , because the frame B is displayed on screen at  $BT_{n+1}$  with which the indexing system indexes the frame B.

Let  $PTS(\alpha)$  and  $PTS(I_n^1)$  denote the PTS value for the first frame  $\alpha$  of a segment  $S_\alpha$  presented at the broadcasting time  $BT_n$  and for the first I-frame since  $BT_n$ , respectively. Then, the time difference  $TD(S_\alpha)$  is defined as:

$$TD(S_\alpha) = PTS(\alpha) - PTS(I_n^1). \quad (1)$$

In Fig. 2(a), the time difference  $TD(S_A)$  for the segment  $S_A$  displayed at  $BT_n$  has a positive value because the PVR will display the video starting from  $I_n^1$  including the segment  $S_A$  as shown in Fig. 2(c). However, the time difference  $TD(S_B)$  for the segment  $S_B$  displayed at  $BT_{n+1}$  has a negative value because the client PVR will display the video starting from the first I-frame  $I_{n+1}^1$  since  $BT_{n+1}$  which results in missing frame B that is desired to be presented as the first frame of the segment  $S_B$ .

Therefore, when we display a segment whose start time is  $BT_n$ , we propose that  $I_{n-1}^1$ , which precedes  $I_n^1$  with a broadcasting time unit ( $BT_n - BT_{n-1}$ : one second in case of STT) from  $BT_n$ , should be used to avoid missing the first frame of the segment we use.



**Fig. 2.** An example of the media localization: (a) The indexing system indexes broadcast stream with the broadcasting time  $BT$ . (b) The generated metadata is described the broadcasting time. (c) The client PVR locates the start position of the segment by the broadcasting time.

## 2.2 Real-Time Segmentation Using Spatiotemporal Visual Pattern

Several approaches [9-11] have recently been proposed for an automatic video indexing by analyzing video, audio and closed caption. However, with the current state of art technology on image understanding and speech recognition, it is still hard to accurately detect highlights and generate a meaningful metadata in real-time.

In order to index a broadcast program in real-time, an operator might have to watch carefully the current broadcast program and manually determine the start and/or end times of events before a broadcast program ends. The event is usually composed of a shot or a set of subsequent shots many of which might be automatically detected by a suitable algorithm with false alarms and missing shots due to editing effects such as zooming in/out, fading, dissolve, and wipe. To get the exact time information of the events, the operator might have to verify the result of automatic algorithm by playing back suspicious segments repeatedly, which will take lots of time. Thus, in order to overcome such problems and quickly index the live broadcast program, we need a new tool for easily verifying shot boundaries.

A spatiotemporal visual pattern called Visual Rhythm [12] also known as spatio-temporal slice [13] provides an efficient way of verifying video segments, which is a two-dimensional abstraction of the entire three-dimensional content of the video.

The most distinguished feature of the visual rhythm is that different video effects including edits and others such as cuts, wipes, zooms and camera motions manifest themselves as different visual patterns on the visual rhythm, as shown in Fig. 3. Due to the features, an operator can find out missing shot boundaries, for example, the wipe in  $shot_n$  in the right side of Fig. 3, which might not be detected by the automatic scene change detection. The operator divides manually the  $shot_n$  into two shots,  $shot_{n1}$  and  $shot_{n2}$ , so as to determine the segment boundary of  $segment_m$  and  $segment_{m+1}$ .

Therefore, inclusion of the visual rhythm in user interface of the real-time indexing application aids an operator to easily and quickly identify segment boundaries as well as visual rhythm itself might be used as a primitive material for automatic shot detection.

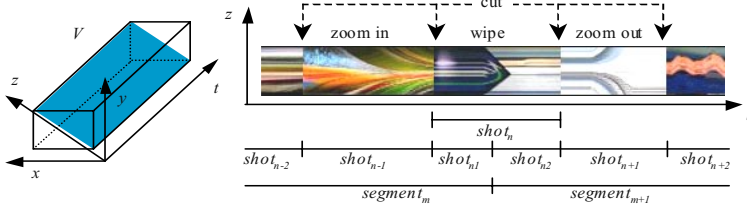


Fig. 3. (a) VR extraction from the video  $V$ . (b) Editing effects presented in VR.

## 2.3 Metadata Delivery

One way to describe segmentation metadata is by utilizing international standards on metadata specification such as MPEG-7 or TV-Anytime. The MPEG-7 or TV-Anytime metadata can be multiplexed into MPEG-2 transport stream that is broadcast to clients through broadcasting network. There might be several solutions of

delivering the standard metadata to clients through broadcast stream: defining a new MPEG-2 private section or descriptor, using the DSM-CC sections, or specifying new type of MPEG-2 PES.

These approaches have two inherent problems. First, the segmentation metadata generated based on the metadata standards are often large in size and thus occupies non-negligible amount of bandwidth for data broadcasting that the current DTV service providers want to minimize. Second, it will take much time for the approaches to be realized because they will require many changes or adoption of existing or new software and hardware components in existing broadcasting environment.

Therefore, a new technique is needed to deliver the segmentation metadata that is smaller in size compared to segmentation metadata based on MPEG-7 and TV-Anytime, through the existing broadcasting environment.

In the proposed system, instead of defining new field for the segmentation metadata, we adopt the existing EPG (Electronic Program Guide) as a carrier of the segmentation metadata because it could be used without any modification of broadcast equipments. That is, we utilize the field for detailed description (synopsis) of a program in EPG data structure. Since the detailed description of a program is presented in the viewer's screen, we have designed new compact metadata format to be legible and informative for viewers who do not have metadata browsing modules only ported on our test-bed client PVR. In table 1, the syntax of the segmentation metadata is represented according to BNF (Bacchus Naur Form) grammar, and one example used in our test-bed is given. The size of the example metadata in table 1 is only 239 bytes whereas the TV-Anytime format for the metadata requires more than 5K bytes for same segmentation information. Due to the small size, we can carry it on the detailed description of a program in EPG which is practically restricted in size of 250 bytes in our test-bed.

**Table 1.** BNF grammar for the our segmentation metadata format and the example.

BNF grammar for metadata format	Example
<pre>&lt;segment_info&gt; ::= &lt;title&gt; &lt;segments&gt; &lt;title&gt; ::= &lt;string&gt; LF &lt;segments&gt; ::= &lt;segment&gt;   &lt;segment&gt; &lt;segments&gt; &lt;segment&gt; ::= &lt;segment_locator&gt; SP [&lt;segment_title&gt;] LF &lt;media_locator&gt; ::= &lt;2digit&gt; ':' &lt;2digit&gt; ':' &lt;2digit&gt; &lt;segment_title&gt; ::= &lt;hierachical_sequence&gt; SP &lt;string&gt; &lt;hierachical_sequence&gt; ::= '*' &lt;sequence_number&gt; '&gt;' &lt;sequence_number&gt; ::= DIGIT   DIGIT '.' &lt;sequence_number&gt; &lt;string&gt; ::= CHAR   CHAR &lt;string&gt; &lt;2digit&gt; ::= DIGIT DIGIT</pre>	<pre>Survival English SEP 4 06:20:41 &lt;1&gt; Introduction 06:22:49 &lt;2&gt; Today's Dialog 06:23:19 &lt;3&gt; Dialog Part I 06:27:04 &lt;4&gt; Dialog Part II 06:31:00 &lt;5&gt; Dialog Part III 06:33:08 &lt;6&gt; More Expressions 06:34:25 &lt;7&gt; Review Dialog 06:35:48 &lt;8&gt; Help Me</pre>

### 3 Implementation and Experimental Results

Real experiments with ATSC terrestrial HDTV programs are performed by porting our software into a commercially available PVR. The scenario we have implemented is as follows. Firstly, we index a broadcast program in real-time and immediately send the resulting metadata to a broadcast station.

Secondly, the delivered metadata of the program is inserted into the field for synopsis of the program in EPG that is transmitted to client PVRs through the broadcasting network.

Finally, the client PVR detects the EPG update and retrieves the metadata of the program in the delivered EPG. The client PVR then locates a segment of interest from the recorded stream according to the broadcasting time described in the delivered metadata. Thus, the client PVR user can browse the recorded program through functionalities such as segment play/replay and random access to the segment of interest.



**Fig. 4.** The timestamp and the real-time indexer using spatiotemporal visual pattern.

The RTIS is composed of a real-time indexer (personal computer) equipped with an encoder for low bit-rate encoding, and a timestamp mixer that is a STB including timestamp generator. We implemented the timestamp mixer by programming the timestamp generator module and then porting it onto the commercially available PVR. Fig. 4 shows the example of the timestamp represented with structured color-code [8] superimposed onto the frame, and the screen shot of indexing application that indexes broadcast program in real-time using the visual timeline called the visual rhythm shown in the top of the application.

For the client PVR in our test-bed, we have utilized a commercially available PVR that is a HDTV STB with a 40G Bytes of HDD, on which we developed our applications. One application is responsible for retrieving the metadata contained in the EPG: checking the EPG update, extracting the metadata of a recorded program from the EPG, and storing the metadata onto the storage. The other application is related with browsing the recorded program with the retrieved metadata: locating a video segment of interest, extracting key frames (thumbnail images) which are used for user interface for browsing window, and managing graphic user interface.

For the experiments, we indexed an educational program that was broadcast at 6:20 AM in Korea. We indexed the program while it was being broadcast using the real-time indexer as shown in the right side of Fig. 4. The indexing process was finished at a minute after ending time of the program. We manually verified the segmentation results, and then generated the metadata such as shown in Table 1. Immediately after generating the segmentation metadata, we sent the metadata through email to an operator who is responsible for updating EPG of a broadcaster. The operator then updated the detailed description (synopsis) of the program using EPG builder with the received metadata. It took some minutes because the operator had to check his email and copy the metadata script and then paste it on the input field of the detailed description of the program manually. Finally, after applying the EPG update in the broadcaster, the metadata was transmitted or broadcast through the broadcast network

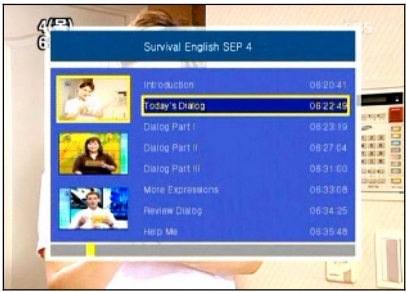


Fig. 5. The graphic user interface of the browser in PVR client.

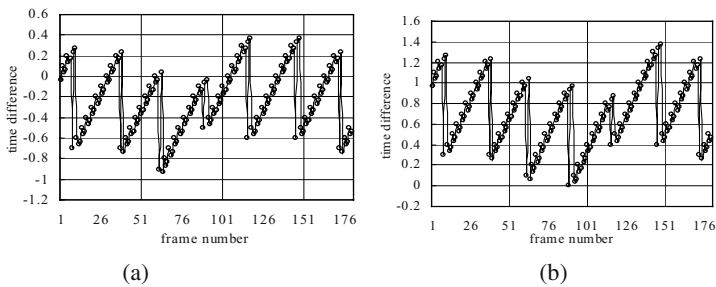


Fig. 6. (a) The time difference by (1). (b) The time difference of proposed method.

and finally received by the client PVR that eventually extracted the metadata. In the experiment, it took about 5 minuts from the ending time of the program to the time of receiving the metadata on the client PVR. This time delay is mainly due to the manual works for sending an email and updating EPG. If we have an interactive channel between the EPG builder and our real-time indexer and the update of EPG can be controlled by software, we could reduce most of the time delay. Thus, PVR users can browse a recorded program with corresponding metadata just after the recording is finished.

Fig. 5 shows the resulting TV screen displayed in PVR when we browse the recorded program with the delivered metadata. The key frame shown in the left of the screen is the image extracted from the recorded stream in PVR by using the broadcasting time described in the delivered metadata.

In our experiment, we observed that the first part of a segment was often missed. To see how much time difference was occurred, we measured the time difference (1) with broadcast stream as shown in Fig. 6(a). Negative values of the time differences in Fig. 6(a) indicate that the video was started playing after the absolute time difference from the desired starting time position of the segment. On the other hand, the time difference of the proposed scheme has no negative value as shown in Fig. 6(b) since we subtracted one second (based on ATSC STT) from the broadcasting time described in the metadata to avoid missing frames when we implemented the browsing module onto the PVR.



## 4 Conclusion

We have introduced a real-time metadata service scheme and implemented a test-bed having an indexing host, a broadcaster, and a client PVR. For the service scheme, we have proposed a novel method of indexing the broadcasting program in real-time, which is to utilize broadcasting time that is carried on the broadcast stream itself. From the experiments, we could show that the method could be applied to the current digital broadcast environments without changing any software and hardware components. Moreover, it was very beneficial demonstration for digital broadcasting, in the point of real-time metadata service for live broadcast program.

**Acknowledgments.** We would like to appreciate Educational Broadcasting System in Korea and Samsung Electronics Co., LTD for allowing us to use their equipments.

## References

1. ISO/IEC 15938-5 Int. Standard Information Technology – Multimedia content description interface – Part 5 Multimedia Description Schemes. ISO/IEC JTC1/SC29/WG11 (2002)
2. TV-Anytime Forum SP003v13 Metadata Specification Version 1.3. TV-Anytime Forum Specification Series: S-3 (Normative). TV-Anytime Forum (2003)
3. ISO/IEC 13818-6 Int. Standard Information Technology – Generic coding of moving pictures and associated audio information: Digital Storage Media Command and Control. ISO/IEC JTC1/SC29/WG11 (1998)
4. ETSI/EBU TS 102 812 V.1.1.1 Digital Video Broadcasting (DVB): Multimedia Home Platform (MHP) Specification 1.1. ETSI. (b2001)
5. A. McPrland, J. Morris, M. Leban, S. Rarnall, A. Hickman, A. Ashley, M. Haataja, F. deJong.: MyTV: A practical implementation of TV-Anytime on DVB and the Internet. International Broadcasting Convention. (2001)
6. ATSC Standard A/65B Program and system information protocol for terrestrial broadcast and cable (Revision B). Advanced Television Systems Committee. (2003)
7. ETSI/EBU EN 300 468 V1.4.1 Digital Video Broadcasting (DVB): Specification for Service Information (SI) in DVB Systems. European Telecommunications Standards Institute (2000)
8. J.-C. Yoon, H. Kim, S. Oh, and S. Sull.: Design of Color-Code System for Time-Stamping Broadcast Video. IEEE Trans. on Consumer Electronics, Vol. 49, No. 3. (2003) 750-758
9. B.T. Truong, S. Venkatesh, and C. Dorai.: Scene Extraction in Motion Pictures. IEEE Trans. on Circuit and Systems for Video Technology, Vol. 13, No. 1. (2003) 5-15
10. S.-C. Chen, M.-L. Shyu, W. Liao, C. Zhang.: Scene change detection by audio and video clues. Proceedings of IEEE ICME 2002, Vol. 2. (2002) 365-368
11. N. Babaguchi, Y. Kawai, and T. Kitahashi.: Event Based Indexing of Broadcasted Sports Video by Intermodal Collaboration. IEEE Trans. on Multimedia, Vol. 4, No. 1. (2002) 68-75
12. H. Kim, J. Lee, J. Yang, S. Sull, W. Kim and S. M. Song.: Visual rhythm and shot verification. Multimedia Tools and Applications, vol. 15. (2001) 227-245
13. C.-W. Ngo; T.-C. Pong; H.-J. Zhang.: Motion analysis and segmentation through spatio-temporal slices processing. IEEE Trans. Image Processing, Vol. 12, Issue 3. (2003) 341-355

# Finding Person X: Correlating Names with Visual Appearances

Jun Yang, Ming-yu Chen, and Alex Hauptmann

School of Computer Science, Carnegie Mellon University  
Pittsburgh, PA 15213, USA

{juny, mychen, alex}@cs.cmu.edu  
<http://www.informedia.cs.cmu.edu>

**Abstract.** People as news subjects carry rich semantics in broadcast news video and therefore finding a named person in the video is a major challenge for video retrieval. This task can be achieved by exploiting the multi-modal information in videos, including transcript, video structure, and visual features. We propose a comprehensive approach for finding specific persons in broadcast news videos by exploring various clues such as names occurred in the transcript, face information, anchor scenes, and most importantly, the timing pattern between names and people. Experiments on the TRECVID 2003 dataset show that our approach achieves high performance.

## 1 Introduction

The dramatic increase of digital videos demands more efficient and accurate access to video content. Content-based analysis and retrieval has been extensively used for video segmentation [2], video retrieval [3], and image retrieval [1]. As discussed in [4], finding a specific person in videos is essential to understand and retrieve videos. Although solving this problem might be difficult for general videos, in this paper we target at very specific content namely broadcast news video. Since news videos are strongly related to human subjects, finding "*person X*" is an important and frequent challenge. Taking advantage of the multimodal content in videos, we propose a people-finding approach which exploits name occurrence in transcript, video structure, and visual information such as faces and news anchor scenes. Specifically, this approach utilizes a timing model to overcome the temporal offset between names and persons, which will otherwise compromise performance. Our approach was developed and evaluated using the dataset from TREC 2003 Video Track (VIDTREC) [5], which is divided into a training set (FSD) and a testing set (FST), each consisting of over 100 hours of ABC, CNN, and C-SPAN news video.

## 2 Transcript Search with Timing-Based Score Propagation

An essential clue for finding a person in the broadcast news video is the mention of his/her name in the transcript, acquired either from a speech recognizer or from closed captions. This clue indicates that this person is likely to appear visually. We do not address the rare cases where a person appears without his/her name being mentioned.

In this section, we discuss using transcript to find and rank video shots that contain specific persons. Here a video shot is defined as an unbroken sequence of frames taken by one camera and it serves as a basic structural unit in our video retrieval.

## 2.1 Basic Transcript-Based Search

Since the transcript is temporally aligned with the video, each shot is associated with a portion of the transcript that falls within its boundary. Therefore, an intuitive way to finding a specific person in video is to use text-based retrieval techniques to find the shots which contain the name. Specifically, we employ the TFIDF retrieval method [6], which gives the similarity between a shot  $S$  and a person named  $X$  as:

$$R(X, S) = \sum_{t_i \in X} tf_i \cdot \log \frac{N}{n_i} \bigg/ \sqrt{\sum_{t_i \in X} \left( tf_i \cdot \log \frac{N}{n_i} \right)^2} \quad (1)$$

where  $tf_i$  is the frequency of term  $t_i$  (as a part of the name  $X$ ) in the transcript of shot  $S$ ,  $N$  is the total number of shots, and  $n_i$  is the number of shots whose transcript has  $t_i$ .

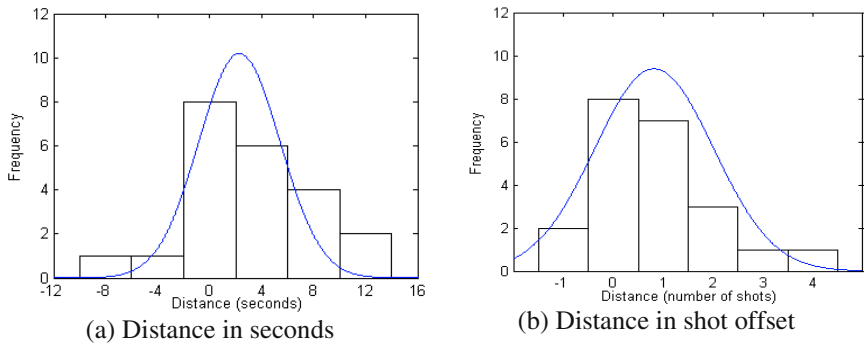
## 2.2 Modeling Timing Between Names and Persons

The method above is subject to a severe problem: it is not necessarily the case that a person appears in the video concurrently with the name mentioned in the transcript. Based on the statistics we have collected, in more than half the cases, a person does not show up in the shot where the name is mentioned, but before or after that shot. Undoubtedly, this mismatch seriously compromises the performance of text-based shot retrieval, which explores only the shots containing the person's name.

The timing between visual appearances (i.e., face) and occurrences of a name is related to the "video grammar" of broadcast news. In a typical news story, an anchorperson briefs the news at the beginning, followed by several shots showing the news event and sometimes interviews and reporters. The name of a human subject in the news is normally first mentioned by the anchorperson, while his/her face is not always shown at that time. In the following shots, this person may appear several times in the video, roughly interleaved with occurrences of the name in the transcript. However, there are also cases where a person not mentioned by the anchorperson later appears in the shots, with or without his name mentioned in close proximity.

Generally, no simple pattern is able to capture the possibility of such timing, but it is still true that a person is more likely to appear in the (temporal) proximity where his name is mentioned. Loosely speaking, the closer is the shot to name occurrence, the more likely it contains the person's visual appearance. As an example, we collected all the visual appearances of "Bill Gates" in FSD, and plot in Fig.1 the frequency of these appearances at each quantized distance from their closest occurrence of his name. The distance is measured in terms of time or shot offset (number of shots between). The "0" point on the distance axis is where the name is mentioned, and positive distance means that a person appears visually after the name is mentioned.

Based on Fig.1, it is intuitive to model the frequency of a person's visual appearance w.r.t his name occurrence using a Gaussian model. For a specific person, we estimate a Gaussian distribution from the distances from each of his visual



**Fig. 1.** The frequency of Bill Gates' visual appearances associates with name occurrences, and the Gaussian curves capturing the frequency distribution.

appearances in FSD to the *closest* name occurrence, both of which are manually labeled, using maximum likelihood estimation. Again, the distance is measured in terms of time or shot offset. In Fig.1, we superimpose the curves of the estimated Gaussian distributions for "Bill Gates", which nicely capture the shape of the bins showing the frequencies.

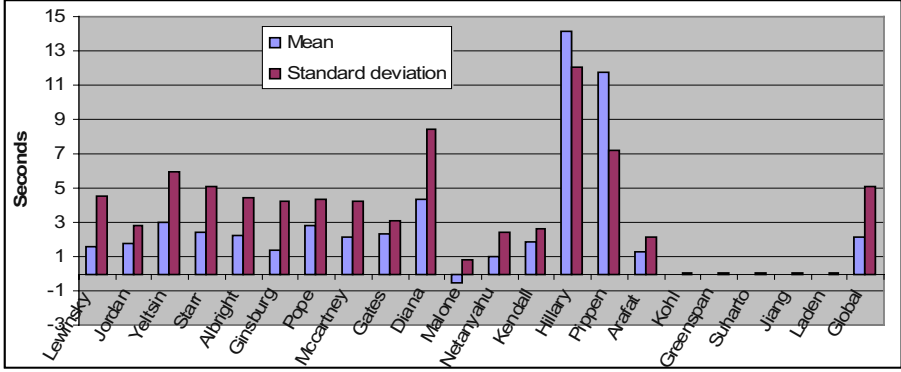
Totally 20 persons are selected for study, varying from frequently appearing ones like "Michael Jordan" to rare ones like "Alan Greenspan". Table 1 shows the number of visual appearances of each person in FSD and FST respectively. The mean and standard deviation of the Gaussian distribution of each person estimated on FSD is plotted in Fig.2 (a) for time-based distance and in Fig.3 (b) for shot-based distance. People are ordered from left to right in descending frequency of their visual appearance in FSD. A global distribution computed from a pool of the training data from all the people is shown alongside.

**Table 1.** The 20 people studied and the number of their visual apperances in FST and FSD.

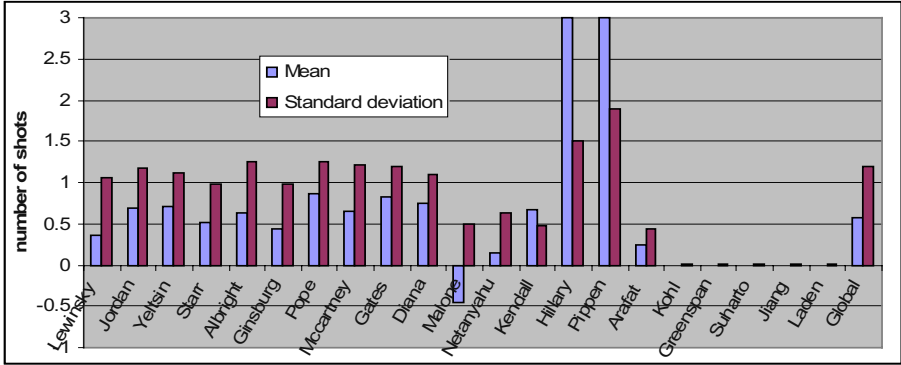
Name	Lewinsky	Jordan	Yeltsin	Starr	Albright	Ginsburg	Pope	Mccartney	Gates	Diana
FSD/FST	53 / 44	47 / 75	40 / 10	37 / 35	30 / 40	28 / 22	29 / 45	26 / 10	22 / 19	12 / 7
Name	Malone	Netanyahu	Kendall	Hillary	Arafat	Kohl	Greenspan	Suharto	Jiang	Laden
FSD/FST	11 / 19	7 / 42	6 / 3	6 / 12	3 / 33	3 / 6	2 / 6	2 / 20	2 / 19	0 / 26

As shown in Fig.2 (a), for the first 9 people on the left, each of who appears 20+ times in FSD, the estimated distributions have similar mean values (1-3 sec.) and moderate standard deviations (3-6 sec.). This suggests that the Gaussian assumption is reasonably good for these people, and their distributions are similar to each other. Therefore, on average a person appears about 2 seconds after his name is mentioned in the "grammar" of news video. For the people with less than 20 appearances in FSD, however, the estimated distributions differ significantly: the mean varies from -2 to 14 seconds, and the standard deviation can be as large as 12 seconds. But it is not fair to say that each infrequent name has a unique distribution, since our observation is biased by the insufficient training data in FSD used to estimate their distributions. We

will explore this question further in our experiments. The same trend is observed in the shot-based distributions in Fig.2 (b).



(a) time-based distance



(b) shot-based distance

**Fig. 2.** The mean and standard deviation of the Gaussian distributions for each person

### 2.3 Search Methods with Score Propagation

Given the timing information, it is obvious that the basic transcript-based search can be improved by propagating the similarity scores from the shots containing the intended person's name to the neighboring shots in a window. The propagation is carried out as:

$$R_p(X, S) = \sum_{|S-S_i| < w} f(S, S_i) R(X, S_i) \quad (2)$$

where  $w$  is the size of the window measured either by time or by shot offset, and  $f(S, S_i)$  is a weighting function with output within  $(0, 1)$ , which decides the score being propagated to neighboring shots. The summation traverses all the shots  $S_i$  that are in the neighborhood of  $S$  and have the intended name in the transcript.

The weighting function  $f(S, S_i)$  can take many forms, depending on the design decisions made along the following dimensions:

- *Flat window or weighted (Gaussian) window*: In a flat window,  $f(S, S_i)$  is a constant and all the shots in the window are propagated with the same score. In a weighted window, however, the score propagated to each shot is determined by its probability of containing the person's visual appearance, which is calculated from the density function of a Gaussian distribution. In this case,  $f(S, S_i)$  is

$$f(S, S_i) = \int_{start}^{end} N(u, \sigma^2) \quad (3)$$

where *start* and *end* are the starting and ending position of  $S$  in relation to  $S_i$  (which has the intended name), and  $N(u, \sigma^2)$  is the density function of the Gaussian distribution.

- *Time-based or shot-based distance measure*: This decides whether to use a time-based Gaussian model  $N_X^t(u, \sigma^2)$  or a shot-based one  $N_X^s(u, \sigma^2)$ . This makes a difference since the shot length differs a lot, and it is unclear which measure is more desirable as to revealing the relationship between a person's visual appearance and the name occurrence.
- *Local, global or combined Gaussian distribution*: To search for a person, we can use the local Gaussian distribution trained particularly for this person  $N_X(u, \sigma^2)$ , the global distribution trained on all the people  $N_G(u, \sigma^2)$ , or a combination of them  $N_C(u, \sigma^2)$ . Intuitively, if each person has a unique distribution and there is enough training data, the local (people-specific) model is more desirable; otherwise the global one is better. The combined model uses a distribution integrated from both the local distribution and the global one. Inspired by the smoothing techniques used to overcome the sparse training data problem in information retrieval [8], this model "smoothes" a person's local distribution estimated from insufficient data with the global distribution. Specifically, the probability density function of the combined distribution is a linear combination of that of the local and the global distribution, where the weight is determined by the amount of training data associated with the person. It is formulated as:

$$N_C = \alpha N_X + (1 - \alpha) N_G \quad \text{and} \quad \alpha = \text{sigmoid}\left(\frac{T_X}{\beta} - \gamma\right) \quad (4)$$

where  $\alpha$  is the weight computed from the number of training data  $T_X$  for person  $X$ , and  $\beta$  and  $\gamma$  are constants, which are set to 10 and 1 as determined by our informal experiments. According to the property of sigmoid function,  $\alpha$  approaches 1 when  $T_X$  increases, and vice versa (e.g.,  $\alpha = 0.5$  when  $T_X=10$ , and  $\alpha = 0.88$  when  $T_X=30$ ). Therefore, the more training data we have observed, the more the combined distribution is determined by the local distribution.

### 3 Face Searching and Anchor Filtering

Visual information provides valuable clues for finding a person in news video. Unlike text information which roughly estimates where a person is, visual information can tell the exact position and time of the person's appearance. Face recognition

technology can match a person's face visually and predict its identity, though its performance is significantly affected by pose and illumination variances. Another important visual clue comes from the anchor detection, since people as news subjects seldom occur during the anchor shot.

We apply the well-known Eigenface algorithm [9] for face recognition. Faces are collected using a face detection system [10], converted to gray levels and normalized to a standard size. Principal component analysis (PCA) is performed to construct Eigenfaces, which encode the most distinguishing parts of faces while ignore similar parts. The Eigenface representation has been shown to be a fairly robust approach to face recognition. However, it also has several drawbacks and the most serious one is pose variations, as non-frontal faces usually have much poorer recognition results than frontal ones. Lighting conditions present another serious problem. In broadcast news, due to the large variations in news footage, both the pose and lighting condition of faces vary largely, resulting in unreliable face recognition.

To avoid the face recognition difficulties, we first use the trustworthy text information to find some shots as initial results, and apply face recognition on them to obtain additional clues for refining the initial results. In this way, the number of faces to be recognized is largely reduced and the accuracy can be improved. To address the wide variance on pose and lighting conditions, we find external images that contain the target face with varied conditions and use them as examples to recognize relevant faces. The (internal) faces to be recognized are extracted from the i-frame of the shots to be examined. Let the external Eigenfaces be denoted as  $\{F1, F2, F3 \dots, Fn\}$  and the internal Eigenfaces be denoted as  $\{f1, f2, f3 \dots, fm\}$ . By matching every internal face with a specific external face  $F_j$  based on Eigenface, we obtain a ranking of all internal faces ordered by descending similarity to  $F_j$ . The final rank of an internal face is combined from its ranks with all the external faces, given as:

$$R(f_i) = \frac{1}{n} \sum_{j=1}^n \frac{1}{R_j(f_i)} \quad (5)$$

where  $R_j(f_i)$  denotes the similarity rank of internal face  $f_i$  with external face  $F_j$ , and  $R(f_i)$  denotes the final rank of  $f_i$ . Since the external faces provide variances in pose and lighting condition, the final rank gives us a more robust prediction. Since a shot may has more than one i-frames, we average the rank of the face on every i-frame of the shot to get the score indicating how likely the shot contains the target face. More details of our face recognition method can be found in [11].

The inclusion of anchor detection assumes that anchors seldom co-occur with a news subject person. We have built an anchor detector [3] based on multimodal classification that combines three information sources: the color histogram from image data, speaker ID from audio data, and face information from face detection. Face information contains the position, size and detection confidence of faces. Fisher's Linear Discriminant (FLD) is applied to select distinguishing features for each source of information. Selected features are synthesized into a new feature vector of each shot, and the classification is performed on these feature vectors.

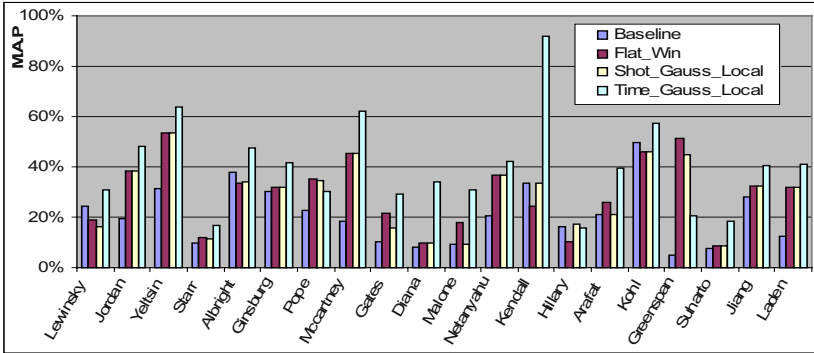
The final prediction of the appearance of the target person is made by linearly combing the results of text-based search, anchor detection and face recognition:

$$P(S) = \alpha T_{prior}(S) + \beta Anchor(S) + \gamma F(S) \quad (6)$$

where  $\alpha$ ,  $\beta$  and  $\gamma$  are weights for the three predictions, which are trained on a held-out set from FST (as FSD has been used to train to distribution).

## 4 Experiment Results

Experiments in finding the 20 selected persons in the TRECVID 2003 collection are conducted to determine the best people-finding method among those proposed in Sect. 2.3. Firstly, we compare the performance of the basic transcript-based search method without score propagation (denoted as *Baseline*), the method with flat-window propagation (*Flat\_Win*), the one with shot-based Gaussian propagation using the local distribution estimated from FSD (*Shot\_Gauss\_Local*), and its time-based counterpart (*Time\_Gauss\_Local*). For each person, we use each method to find the shots in FST that contain his/her visual appearance and compute the *mean average precision* (MAP) [7] of the results. Note that the propagation window sizes in each method have been fine-tuned based on the FSD data.



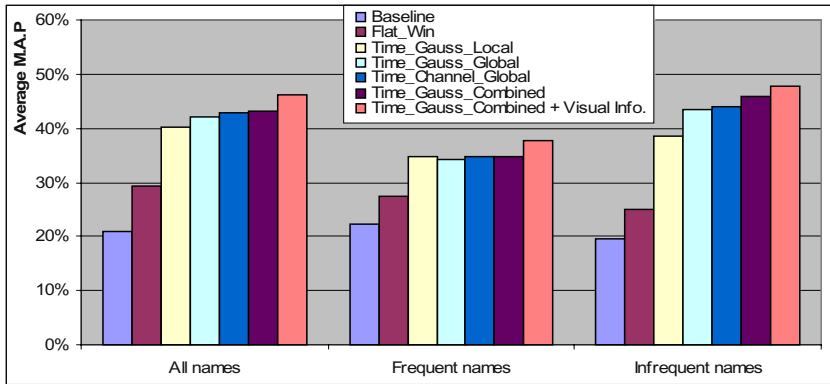
**Fig. 3.** Performance comparison of three propagation methods with baseline method

As shown in Fig.3, in all the 20 queries at least one propagation approach outperforms the baseline, and for 15 queries among them, all the three propagation approaches outperform the baseline. This suggests that score propagation based on timing information can greatly help the task of people-finding. Moreover, in 17 out of the 20 queries, the time-based Gaussian approach is the best performer, whose average MAP (0.40) is much higher than that of the flat-window approach (0.29) and shot-based Gaussian approach (0.28). Thus, time-based Gaussian is a better propagation strategy than the other two, implying that time is a better distance measure than shot offset w.r.t. revealing the timing between names and people.

Fig. 4 shows the average MAP (over 20 queries) of the time-based Gaussian method using local, global, and combined distribution respectively, in comparison to that of baseline and flat-window approach. As shown, the approach with combined distribution outperforms the global one by 2%, which beats the local one by another 2%, and all are about twice the performance of the baseline approach.

The three types of distribution cause more interesting discrepancy on the performance of finding frequently occurring people versus that of finding infrequent





**Fig. 4.** Performance comparison of local, global, combined distribution with visual information

ones. Here frequent people are those who appear visually 20+ times in both FSD and FST (cf. Table 1), while infrequent ones are those appearing 20- times in both FSD and FST. By this standard, there are 7 frequent and 8 infrequent people among the 20 people, while 5 people cannot be clearly classified due to their unbalanced appearances in FST and FSD. As we can see, for frequent names the choice of distributions does not have any significant influence on the performance, while for infrequent names the difference is substantial. Specifically, for all the 7 frequent people, the MAP of global distribution never differs from that of local distribution by over 10%, while for 5 out of the 8 infrequent ones, global distribution enhances the MAP by over 20%. This echoes our observation in Sect.2.1 that the distribution of frequent names is similar to each other and thus to the global one, which is dominated by the dense training data of frequent people. Therefore, the performance of finding such people is almost unaffected by the choice of distribution. For infrequent people, since their local distribution is poorly estimated using their insufficient training data, the performance can benefit from using the more stable global distribution. It is interesting to see that the combined distribution is better than the global one, which implies that each name has a unique "true" distribution that lies between the global and the local one. However, this conclusion can be challenged due to insufficient queries (8 infrequent names) and the small improvement (about 4%).

Since our data consist both ABC and CNN news, it is interesting to know if these two channels have different styles that lead to different distributions. Thus, we train two channel-specific global distributions on FSD and test them on FST. As shown in Fig.4, this approach (*Time\_Channel\_Global*) improves MAP over the uniform global distribution by only 1%, suggesting that ABC and CNN have similar editing styles.

Finally, we combine transcript search with time-based smoothed distribution and vision information. The combination weights we trained from the held-out set are 1.0 for transcript information, -0.812 for anchor filtering and 0.087 for face recognition. These weights reflect the fact that face recognition is very unreliable, while the anchor detection has the ability to remove false positives. As shown in Fig.4, combining transcript with visual information gave another 3% improvement, which is mainly derived from anchor detection. Among the 20 people, the visual information enhances the MAP on 4 people substantially (over 20%), and we find that they all

appear with frontal faces in the video. 10 people have minor improvement (1%-20%) on their MAP with visual information, while the rest 6 people do not improve at all.

## 5 Conclusion

In this paper, we address the task of finding a person using clues including transcript, video structure, and vision information. Gaussian distribution has been proved experimentally an effective model to describe the timing pattern between a person's visual appearances and the occurrences of his/her names. Specifically, a "smoothed" Gaussian distribution estimated using both the local and global training data produces the best performance, especially for infrequently appearing people. Finally, combining visual information such as face recognition and anchor detection with transcript information brings additional benefit to the person-finding task.

**Acknowledgement.** This research is partially supported by the Advanced Research and Development Activity (ARDA) under contract # MDA908-00-C-0037 and MDA904-02-C-0451.

## References

1. Smeulders, et al.: Content-Based Image Retrieval at the End of the Early Years. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol 22, No 12 (2000) 1349-1379.
2. Zhang, H.J, Kankanhalli, A., Smoliar, S.W. "Automatic partitioning of full-motion video", ACM Multimedia Systems, 1(1), 1993.
3. Hauptmann, A., et al. Informedia at TRECVID 2003: Analyzing and Searching Broadcast News Video, Proceedings of TREC 2003, (2003).
4. Satoh, S. and Kanade, K.: NAME-IT: Association of Face and Name in Video. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (1997) 775-781.
5. The NIST TREC Video Retrieval Evaluation, <http://www-nlpir.nist.gov/projects/trecvid/>.
6. Salton, G., McGill, M.: Introduction to Modern Information Retrieval. McGraw-Hill, New-York (1983).
7. Baeza-Yates, R. and Ribeiro-Neto, N.: Modern Information Retrieval. Addison Wesley, Essex, England (1999).
8. Zhai, C. and Lafferty, J.: A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. Proc. 24th Int'l ACM SIGIR Conf. (2001): pp. 334-342.
9. Pentland, A., Moghaddam, B., and Starne, r T.: View-Based and Modular Eigenspaces for Face Recognition IEEE Conference on Computer Vision & Pattern Recognition (1994).
10. Schneiderman, H. and Kanade T., "Object Detection Using the Statistics of Parts," International Journal of Computer Vision 2003.
11. Chen, M.Y., Hauptmann, A., "Searching for a Specific Person in Broadcast News Video," Int'l Conf. on Acoustics, Speech, and Signal Processing, May, 2004 (to appear).

# A Framework for Semantic Classification of Scenes Using Finite State Machines

Yun Zhai<sup>1</sup>, Zeeshan Rasheed<sup>2</sup>, and Mubarak Shah<sup>1</sup>

<sup>1</sup> School of Computer Science, University of Central Florida  
Orlando, Florida 32828, United States  
{yzhai, shah}@cs.ucf.edu

<sup>2</sup> ObjectVideo  
11600 Sunrise Valley Dr. Suite 290  
Reston, Virginia 20191, United States  
zrasheed@objectvideo.com

**Abstract.** We address the problem of classifying scenes from feature films into semantic categories and propose a robust framework for this problem. We propose that the Finite State Machines (FSM) are suitable for detecting and classifying scenes and demonstrate their usage for three types of movie scenes; conversation, suspense and action. Our framework utilizes the structural information of the scenes together with the low and mid-level features. Low level features of video including *motion* and *audio energy* and a mid-level feature, face detection, are used in our approach. The transitions of the FSMs are determined by the features of each shot in the scene. Our FSMs have been experimented on over 60 clips and convincing results have been achieved.

## 1 Introduction

Recent years have seen a growing interest in the annotation and retrieval of video data. The increasing number of subscribers to digital cable now demands efficient tools so that viewers can browse and search sections of interest of video. Among many genres of video production, feature films are a vital field for the application of such tools. It is a sizeable element of the entertainment industry, easily available, widely watched and therefore, is becoming a focus of researchers in many aspects. For example, applications for content-based video annotation and retrieval have been developed at all levels of the video structure; shot level, scene level, and movie level. A shot is a sequence of images that preserve consistent background settings. It is the basic element of a movie. A scene, which consists of a set of continuous shots, constitutes a portion of the story line. On the highest level, a movie is composed of a series of related scenes defining a theme. For a user, who may be looking for a particular scene of a feature film, a shot level analysis is insufficient since a shot level analysis fails to capture the semantics of the video content. For example, how does one answer a query for a *suspense* scene in a feature film based on a single shot content? Any semantic category like *suspense* or *tragedy*, cannot be defined over a single shot. These concepts are induced in viewers over time. Indeed, a meaningful result can only be achieved by exploiting the interconnections of shot content.

In this paper, we present a novel framework for classifying scenes, focusing on feature films, into three semantic categories; conversation, suspense and action. This method analyzes the structural information of the scenes based on the low-level and mid-level shot features which are robust and easily computable. The low-level features used in our framework include shot motion and audio energy and the mid-level feature is face identity. To bridge the gap between the low and mid-level features and a high-level semantic category, Finite State Machines are studied and developed. The transitions are determined based on the statistics of these features for each shot. This paper is organized as follows: Related work is discussed in Section 2, Section 3 describes the classification framework, including the features and the Finite State Machines for detecting conversation, suspense and action scenes. Section 4 shows the experimental results and Section 5 concludes our work.

## 2 Related Work

In the area of higher level scene understanding, Adams et al [1] proposed the detection of “tempo” in movies. The camera motion magnitude and the shot length were the two features used to compute a continuous function. Our framework, on the other hand, analyzes the structure of the movie scene and classify scenes into more specific categories. Yoshitaka et al. [3] also used shot length and visual dynamics to analyze scene type. In their approach, the color statistics of the frames in the shot were used to calculate the visual dynamics and the similarities between the repeating shots were exploited. Experiments on only one kind of scene was demonstrated and it was not clear how the approach could be extended to other scene categories.

Lienhart et al. [4] used face detection in the scenes to link similar shots. A “face-based class” with a group of related frames showing the same actor was constructed by the similarity of the spatial positions and sizes of the detected faces. These “face-based classes” were linked across shots in the video to form the “face-based sets” by using Eigenfaces. The pattern of a dialog scene was flagged if several conditions were satisfied. In their experiment, face recognition suffered accuracy and the system typically split the same actor into different sets causing over detection. Li et al. [5], exploited the global structural information of a scene and built “shot sinks” to classify a scene into one of three scenarios including *two speaker dialog*, *multi-speaker dialog*, and *others*. The overall structure was computed based on the low-level visual features, such as color of the shots in the scene. In their approach face information, which is an important cue for speaker detection, was not used. We combine both structure and face detection in a Finite State Machine framework to provide a more general solution for the scene classification task.

## 3 Proposed Approach

In this section, we first discuss the low-level and mid-level features used in our approach. The *activity intensity*, which is a function of low-level features and includes local and global motion and audio signal, is the input to the Finite State Machines. Human faces are detected in shots, clustered, and also used as input. We construct FSMs for three different semantic categories of scenes. These include conversational, suspense and action.

### 3.1 Computing Activity Intensity ( $I$ ) Using Motion and Audio

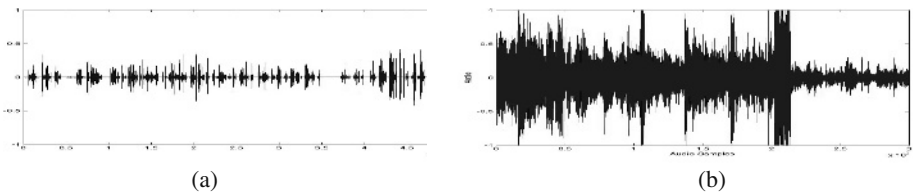
Motion in the videos has been used by several researchers in detecting and identifying scenes in feature films. Some examples are [1,2]. In feature films, the camera motion is generally translation and zoom, whereas, camera roll and tilt are rare. Affine motion model is suitable for capturing translation, scale and rotation about the optical axis of camera. Therefore, we model the image-to-image global transformation using an affine motion model. We exploit the motion vector information embedded in the MPEG compressed video. The approximate motion model is computed based on the 16x16 pixel macro-blocks. For each macro-block  $[x \ y]^T$ , its motion  $[u \ v]^T$  is computed as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & b_1 \\ a_3 & a_4 & b_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (1)$$

where  $[b_1 \ b_2]^T$  vector captures the global translation. The magnitude  $m$  of the translation vector represents the intensity of the motion, and its absolute difference  $d$  across adjacent frames gives the smoothness of the camera motion. Thus, an average global motion quantity,  $\lambda$  over the entire shot captures both intensity and the smoothness of the global motion, that is:

$$\lambda = (m + \kappa_m) \times (d + \kappa_d), \quad (2)$$

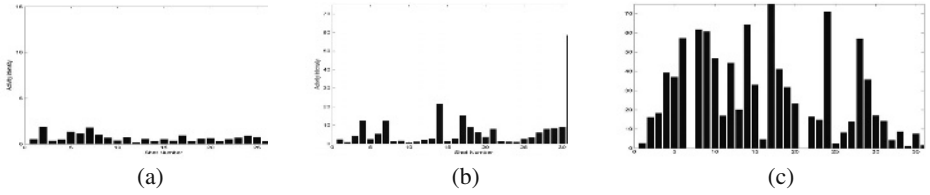
where  $\kappa_m$  and  $\kappa_d$  are small positive constants to avoid multiplication with zero. The local motion intensity also varies with the type of scene. For example, in a fighting scene, high action intensity is commonly observed. We compute the local motion intensity by computing the mean difference,  $\mu$ , of the reprojected motion vectors and the original motion vectors for the entire shot.



**Fig. 1.** The audio signal for (a) conversation, and (b) action.

Sound also plays an important role in distinguishing scenes from each other. In conversational scenes, characters speak smoothly and calmly. In action scenes, which often include explosions, collisions, or vehicle chases, the audio energy is very high. Figure 1 shows the plot of audio signals for (a) conversational and (b) action scenes. Note that the high energy in the audio of the action scene is distinctive from that of the conversational scene. Therefore, the computation of activity intensity also incorporates the mean audio energy  $\theta$ . The overall activity intensity is the combination of the three quantities,  $\lambda$ ,  $\mu$  and  $\theta$  as follows:

$$I = \lambda \times (\mu + 1) \times (\theta + 1) \quad (3)$$



**Fig. 2.** The activity  $\lambda$  of three types of movie scenes (a) conversation, (b) suspense, and (c) action. The horizontal axis represents the shot number in the scene.

Figure 2 shows the histogram of the activity intensity values for three types of shots: (a) conversation, (b) action and (c) suspense.

### 3.2 Face Detection

Conversational scenes generally have shots with at least two humans. We utilize this cue and detect human faces in the video using the method proposed by Viola et al. [6]. We have found that [6] performs reasonably good for faces with different scales in the video. The shots containing faces are clustered together based on a 24-bin RGB histogram. Figure 3 shows human faces in some shots.



**Fig. 3.** Results of face detection in a scene.

### 3.3 Finite State Machines (FSM)

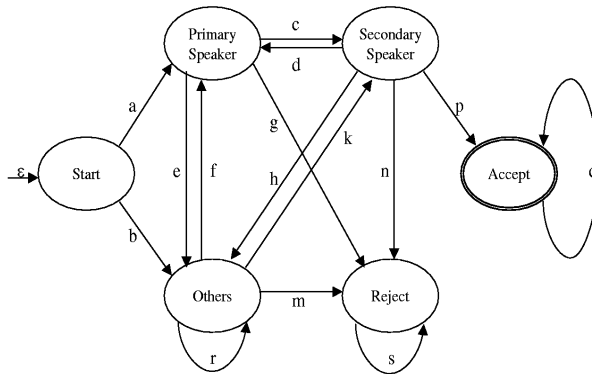
A Finite State Machine is defined as,

$$A = (Q, \Sigma, \sigma, q_0, F), \quad (4)$$

where  $Q$  is a set of states in the FSM and  $\sigma$  is the set of transitions.  $\Sigma$  contains the conditions for the transitions.  $q_0$  is the initial state, and  $F$  is the set of accepting (final) state. In feature films, scenes are generally composed in accordance with the conventional film grammar. We have observed the following characteristics for three different categories of scenes:

- (i) **Conversational scenes:** low activity intensity, medium audio energy and multiple speakers.
- (ii) **Suspense scenes:** a long period of silence followed by a sudden eruption either in sound track or in activity intensity or both.
- (iii) **Action scenes:** intensive action activity for a certain number of shots.

We discuss three different FSMs which detect conversational, suspense and action scenes.



**Fig. 4.** Finite state machine for conversation scene detection. It consists of six states.

**FSM for Conversation Scenes.** Figure 4 shows a deterministic Finite State Machine for detecting conversation scenes. The FSM consists of six states: *Start*, *Primary Speaker*, *Secondary Speaker*, *Others*, *Reject* and *Accept*. Shots with high similarity that contain a face are clustered together. The state *Primary Speaker* is represented by the largest cluster, and the *Secondary Speaker* is represented by the second largest cluster. The transitions are determined based on the feature values of the shots in the scene. If the state *Accept* is reached, the scene is declared as “Conversation” scene. Otherwise it is declared as “Non-Conversation”. In this FSM,  $Q = \{Start, PrimarySpeaker, SecondarySpeaker, Others, Reject, Accept\}$ ,  $q_0 = \{Start\}$  is the initial state and  $F = \{Accept\}$  is the final state. The set of the transitions  $\sigma$  includes  $\{\varepsilon, a, b, c, d, e, f, g, h, k, m, n, p, q, r, s\}$ . The transition matrix for  $\sigma$  is shown in Table 1. The transition conditions  $\Sigma$  are:

- **a:** The first shot in the scene is a facial shot with low activity intensity. Results in the transition to the state *Primary Speaker*.
- **b:** The first shot in the scene is a non-facial shot with low activity intensity. Results in the transition to the state *Others*.
- **d** and **f:** The new shot is a facial shot with low activity intensity, and it belongs to the largest cluster. Results in the transition to the state *Primary Speaker*.
- **c** and **k:** The new shot is a facial shot with low activity intensity, and it belongs to the second largest cluster. Results in the transition to the state *Secondary Speaker*.
- **e, h** and **r:** The new shot is a non-facial shot with low activity intensity or the new shot is a facial shot with low activity intensity but belongs neither to the largest cluster nor the second largest cluster. Results in the transition to the state *Others*.
- **g, n** and **m:** The new shot has high activity intensity. Results in the transition to the state *Reject*.
- **p:** The new shot is a facial shot with low activity intensity. It completes the accepting requirement of the FSM. Results in the transition to *Accept*.
- **q:** For any new shot, this transition loops at the state *Accept*.
- **s:** For any new shot, this transition loops at the state *Reject*.

**Table 1.** Transition matrix for conversation detection. Columns represent “From” states, rows represent “To” states and “-” indicates no transition from one state to another.

$\sigma$	Start	Primary	Secondary	Others	Reject	Accept
Start	-	a	-	b	-	-
Primary	-	-	c	e	g	-
Secondary	-	d	-	h	n	p
Others	-	f	k	r	m	-
Reject	-	-	-	-	s	-
Accept	-	-	-	-	-	q

**FSM for Suspense Scenes.** We have observed that suspense scenes often have the following pattern. In the beginning, the scene is relatively silent and is followed by a sudden increase in sound energy. In many cases, it is also accompanied by abrupt camera and actor movements. Based on these observations, the FSM for detecting the suspense scenes have the following four states: *Start*, *Wait*, *Reject* and *Accept*. The state *Wait* represents the pre-action moments. After a period of *waiting*, the state is transferred to *Accept* if a sudden action shot is seen. The FSM rejects the scenes in which the sudden action happens before the predefined interval.

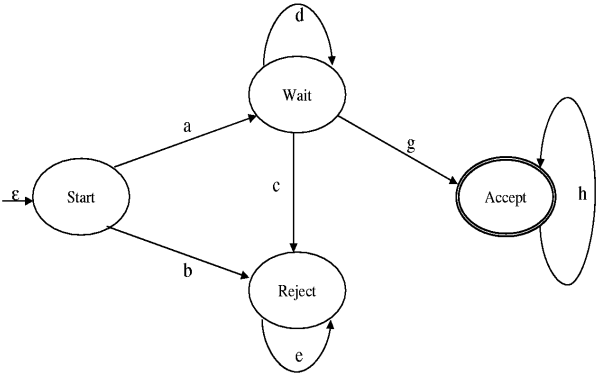
Similarly, the definition of the FSM for the classification of suspense scenes can be written in the general formula for the Finite State Machines. In this case,  $Q = \{Start, Wait, Reject, Accept\}$  are the states. The initial state is  $q_0 = \{Start\}$ , and the final state is  $F = \{Accept\}$ . The transition set  $\sigma$  includes  $\{\varepsilon, a, b, c, d, e, f, h\}$ . The FSM is shown in Figure 5, and the corresponding transition matrix is shown in Table 2. The transition conditions are defined as follows:

- **a:** The first shot in the scene is a shot with low activity intensity. Results in the transition to the state *Wait*.
- **b:** The first shot in the scene is a shot with high activity intensity. Results in the transition to the state *Reject* if the action happens before a predefined time interval.
- **c:** The new shot is a shot with high activity intensity, and the waiting time is less than the required period. Results in a transition to the state *Reject*.
- **d:** The new shot is a shot with low activity intensity, and the waiting time is less than the required period. Loops at the state *Wait*.
- **e:** For any new shot, loops at the state *Reject*.
- **g:** The new shot is a shot with high activity intensity, and the waiting time is more than the required period. Results in the transition to the state *Accept*.
- **h:** For any new shot, loops at the state *Accept*.

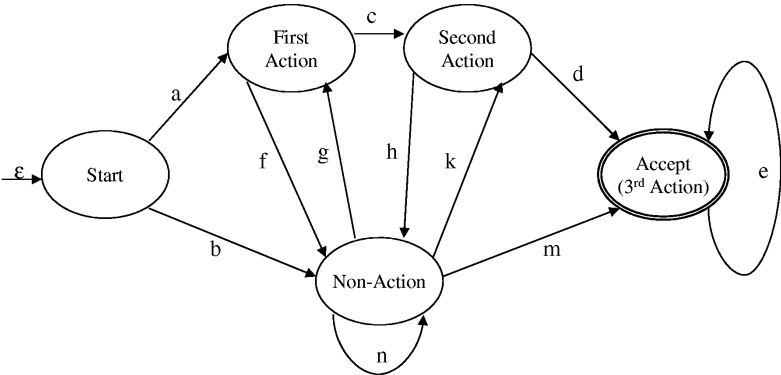
**FSM for Action Scenes.** Action scenes in movies generally have very high action intensity, such as scenes containing explosions, chasing and fighting. To classify a scene as an action scene, the scene must have three or more shots with action intensity higher than a threshold. The FSM for detecting action scenes is shown in Figure

For action FSM, the state set  $Q$  has  $\{Start, FirstAction, SecondAction, Non - Action, Accept(ThirdAction)\}$ , where the initial state  $q_0$  is  $\{Start\}$ , and the final state  $F$  is  $\{Accept(ThirdAction)\}$ . The transition set  $\sigma$  includes





**Fig. 5.** Finite state machine for suspense scene detection. It consists of four states.



**Fig. 6.** Finite state machine for action scene detection. It consists of five states.

**Table 2.** Transition matrix for suspense scene detection. Column represent “From” states, rows represent “To” states and “-” indicates no transition from one state to another.

$\sigma$	Start	Wait	Reject	Accept
Start	-	a	b	-
Wait	-	d	c	g
Reject	-	-	e	-
Accept	-	-	-	h

$\{\varepsilon, a, b, c, d, e, f, g, h, k, m, n\}$ . A *Previous State* attribute for a state  $q_i$  in the FSM and defined as the “from” state of the immediate transition before reaching state  $q_i$ . This is used for the determination of the outgoing transitions from state *Non-Action*. The transition matrix is shown in Table 3. The transition conditions are: 6.

- **a:** The first shot in the scene has high activity intensity. Results in the transition to the *First Action* state.

- **b**: The first shot in the scene has low activity intensity. Results in the transition to the state *Non-Action*. The *Previous State* is set to *Start*.
- **c**: The new shot has high activity intensity. Results in the transition to *Second Action*.
- **d**: The new shot has high activity intensity. Results in the transition to the state *Accept (Third Action)*.
- **e**: For any new shot, loops at *Accept (Third Action)*.
- **f**: The new shot has low activity intensity. Results in the transition to *Non-Action*. The *Previous State* is set to *First Action*.
- **g**: The new shot has high activity intensity. Results in the transition to the state *First Action*. The *Previous State* is set to *Start*.
- **h**: The new shot has low activity intensity. Results in the transition to the state *Non-Action*. The *Previous State* is set to *Second Action*.
- **k**: The new shot has high activity intensity. Results in the transition to the state *Second Action*. The *Previous State* is *First Action*.
- **m**: The new shot has high activity intensity. Results in the transition to the state *Accept (Third Action)*. The *Previous State* is *Second Action*.
- **n**: The new shot has low activity intensity. Loops at *Non-Action*.

**Table 3.** Transition matrix for action scene detection. Column represent “From” states, row represent “To” states and “-” indicates no transition from one state to another.

$\sigma$	Start	1st-Act	2nd-Act	Non-Act	Accept
Start	-	a	-	b	-
1st-Act	-	-	c	f	-
2nd-Act	-	-	-	h	d
Non-Act	-	g	k	n	m
Accept	-	-	-	-	e

4 Experimental Results

We have experimented with over 60 clips using the Finite State Machines for 3 categories of scenes. These clips are taken from 7 Hollywood movies including “The Others”, “Jurasic Park III”, “Terminator II”, “Gone in 60 Seconds”, “Mission Impossible II”, “Dr. No”, and “Scream”. We also included a TV talk show, “Larry King Live” and a TV news program, “CNN Headlines”. The feature movies cover a variety of genres such as horror, drama, and action. Each clip contains approximately 20-30 shots. Four human observers were asked to choose the most suitable label from three categories for each clip. Each clip was given a ground truth label with the category that the most human observers agreed upon. Thus, each clip is considered as a positive member of the category to which it is assigned. Observers were also asked to provide the most unlikely category for each clip. We used this information to label a clip as a non-member (or a negative member) for the unlikely categories.

To evaluate the performance of the proposed approach, two measures of accuracy were computed. These measures are precision and recall and defined as follows:

$$P_{pos} = \frac{M_{pos}}{D_{pos}}, \quad R_{pos} = \frac{M_{pos}}{G_{pos}} \quad (5)$$

and

$$P_{neg} = \frac{M_{neg}}{D_{neg}}, \quad R_{neg} = \frac{M_{neg}}{G_{neg}}, \quad (6)$$

where  $P_{pos}$ ,  $R_{pos}$ ,  $P_{neg}$  and  $R_{neg}$  are the precision and recall for positive and negative member detection.  $G_{pos}$  and  $G_{neg}$  are the ground truth.  $D_{pos}$  and  $D_{neg}$  are the detected positive and negative members.  $M_{pos}$  and  $M_{neg}$  are the numbers of the correctly matched positive and negative members.

There were 27 conversational scenes in the data set. The results achieved were 96.2% precision and 92.6% recall. For the other 25 non-conversational scenes, the precision was 92.0%, and the recall was 95.9%. The number of positive members of the suspense category in the data set was 12, with 15 non-member scenes. The precision and recall for the member detection was 100.0% and 93.8% respectively, and the precision and recall for the non-member clip was 91.7% and 100.0% respectively. In action scenes, we had 21 member clips and 29 non-member clips. The precision and recall for the positive members was 87.0% and 95.2% respectively. The precision and recall for the negative members are 96.3% and 89.7% respectively. The overall performance is summarized in Table 4. These results clearly demonstrate that a finite state machine can detect and classify video scenes into categories. Figure 7 shows some clips with the key frames of the shots in the scene.

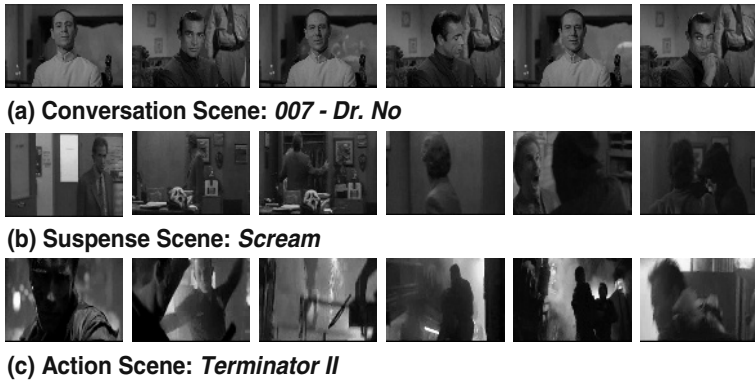


Fig. 7. Three testing clips. The six representative key frames are displayed.

## 5 Conclusions

In this paper, we presented a novel framework for classifying video scenes into high-level semantic categories using deterministic Finite State Machine (FSM). The transitions in

**Table 4.** Precision and recall for conversation, suspense and action scene classification.

<i>Scene Type</i>	<b>Conversation</b>		<b>Suspense</b>		<b>Action</b>	
<i>Accuracy</i>	Positive	Negative	Positive	Negative	Positive	Negative
<b>Precision</b>	96.2%	92.0%	100.0%	93.8%	87.0%	95.2%
<b>Recall</b>	92.6%	95.9%	91.7%	100.0%	96.3%	89.7%

each FSM are based on the low and mid-level shot features. These features are robust and easily computable. We also incorporated face detection to cluster shots and used these clusters to determine the transitions of the FSMs. We demonstrated the usefulness of FSM for this task by experimenting on over 60 movie clips and achieved high recall and precision. In the future, we plan on exploring Finite State Machine to detect scene categories for entire movies.

References

1. B. Adams, C. Dorai, S. Venkatesh, *Novel Approach to Determining Tempo and Dramatic Story Sections in Motion Pictures*, ICIP, 2000.
2. Z. Rasheed, M. Shah, *Scene Detection In Hollywood Movies and TV Shows*, IEEE Computer Vision and Pattern Recognition Conference, Madison, Wisconsin, June 16-22 2003.
3. A. Yoshitaka, T. Ishii, M. Hirakawa, and T. Ichikawa, *Content-Based Retrieval of Video Data by the Grammar of Film*, IEEE Symposium on Visual Languages, 1997.
4. R. Lienhart, S. Pfeiffer, and W. Effelsberg, *Scene Determination Based on Video and Audio Features*, Proc. IEEE Conf. on Multimedia Computing and Systems, Florence, Italy, 1999.
5. Y. Li, S. Narayanan, C.-C. Jay Kuo, *Movie Content Analysis Indexing, and Skimming*, Kluwer Academic Publishers, *Video Mining*, Chapter 5, 2003.
6. P. Viola and M. Jones, *Robust Real-Time Object Detection*, International Journal of Computer Vision, 2001.

# Automated Person Identification in Video

Mark Everingham and Andrew Zisserman

Visual Geometry Group, Department of Engineering Science, University of Oxford  
`{me|az}@robots.ox.ac.uk`

**Abstract.** We describe progress in the automatic detection and identification of humans in video, given a minimal number of labelled faces as training data. This is an extremely challenging problem due to the many sources of variation in a person's imaged appearance: pose variation, scale, illumination, expression, partial occlusion, motion blur, etc.

The method we have developed combines approaches from computer vision, for detection and pose estimation, with those from machine learning for classification. We show that the identity of a target face can be determined by first proposing faces with similar pose, and then classifying the target face as one of the proposed faces or not. Faces at poses differing from those of the training data are rendered using a coarse 3-D model with multiple texture maps. Furthermore, the texture maps of the model can be automatically updated as new poses and expressions are detected. We demonstrate results of detecting three characters in a TV situation comedy.

## 1 Introduction

The objective of this paper is to annotate video with the identities, location within the frame, and pose, of specific people. This requires both detection and recognition of the individuals. Our motivation for this is two fold: firstly, we want to annotate video material, such as situation comedies and feature films, with the principal characters as a first step towards producing a visual description of shots suitable for blind people, e.g. "character A looks at character B and moves towards him". Secondly, we want to add index keys to each frame/shot so that the video is searchable. This enables new functionality such as "intelligent fast forwards", where the video can be chosen to play only shots containing a specific character; and character-based search, where shots containing a set of characters (or not containing certain characters) can easily be obtained.

The methods we are developing are suitable for any video material, including news footage and home videos, but here we present results on detecting characters in an episode of the BBC situation comedy 'Fawlty Towers'. Since some shots are close-ups or contain only face and upper body, we concentrate on detecting and recognizing the face rather than the whole body.

The task is a staggeringly difficult one. We must cope with large changes in scale: faces vary in size from 200 pixels to as little as 15 pixels (i.e. very low resolution), partial occlusion, varying lighting, poor image quality, and motion blur. In a typical episode the face of a principal character (Basil) appears frontal

in one third of the frames, in profile in one third, and from behind in the other third, so we have to deal with a much greater range of pose than is usual in face detection.

Previous approaches to character identification have concentrated on frontal faces [7,9]. This is for two reasons: (i) face detection is now quite mature and successful for frontal faces [10,15,16] (both in terms of false positive/ false negative performance, and also in efficiency); and (ii) because most recognition methods are developed for frontal faces [17]. For example, image-based ‘eigenface’ or ‘Fisherface’ [2] approaches are successful for registered frontal faces with stable illumination. Detection of profile faces [15] or arbitrary pose [10,12] has not yet reached the same level of performance. This is principally because in the case of frontal faces pattern matching methods can be used to classify an image region through a fixed mask as a face or non-face, since there are sufficient distinctive internal features visible (eyes, mouth, etc.). In the case of profiles there are fewer distinctive features, and the silhouette varies. Consequently, simple fixed regions of interest include background, and the resulting learning problem is then much more difficult.

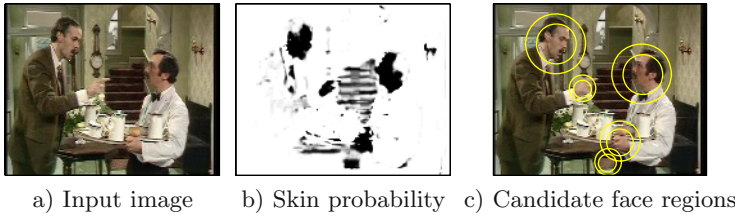
The approach we have developed is closest in spirit to the pose and multiple view based approaches of [3,5,13]. Suppose that we have identified a face region in a target frame, and our task is now to decide if this is the face of one of the characters in our training data. This is a matching problem, and in the case of faces we must account for three principal ‘dimensions’ of variation: pose change, illumination change, and expression change. Conceptually we divide this problem into two parts:

1. *Pose based rendering*: a set of candidate faces is proposed by rendering faces from the training data at the same pose as the target face, see figure 4. The candidate faces will typically contain several examples of the correct face with a range of expressions, as well as examples of other characters. This largely eliminates the pose variation, and we have reduced the problem to matching over expression and illumination change.

2. *Classification*: a matching decision is made amongst the proposed faces. The outcome is a match with one of the faces, or a non-match (if the target face is not one of the learnt characters). This requires a matching measure which is tolerant to small changes of expression, and largely invariant to illumination conditions.

## 2 Approach

In this section we describe the two stages of the algorithm: learning face models, and recognition of faces in target frames. The overall recognition approach consists of three steps: (i) detecting candidate face regions in the target frame, (ii) determining the pose of the target face and proposing candidate faces at that pose, and (iii) classification.



**Fig. 1.** Candidate face region detection using skin colour model and multi-scale blob detector. Darker grey levels in (b) represent higher probability. Concentric circles in (c) show the scale uncertainty in the detections. Note there are several false positives due to non-face skin regions, and non-skin regions of similar colour. These false positives will be removed by subsequent verification.

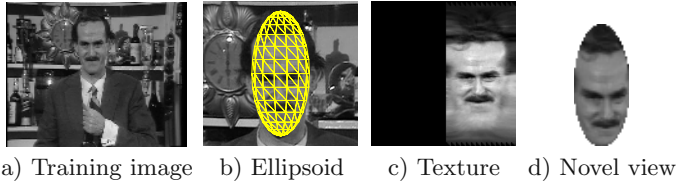
## 2.1 Candidate Face Region Detection

The first step in detection is to propose candidate face regions in an image for further processing. Requirements are that the algorithm proposes all faces in the image as candidates across a wide range of scale and pose. We desire to have a relatively small number of false positive (non-face) responses from the algorithm, since processing false detections incurs computational cost, but we can cope with some false positives since candidate regions will be subsequently verified. This differs somewhat from the isolated problem of face detection [15, 16], where detections are not subject to additional verification.

We take advantage of working with colour video and use a skin colour detector to propose probable face regions. The probability distribution over the colour of skin pixels in RGB space is modelled as a single Gaussian with full covariance. A corresponding Gaussian distribution with large variance is estimated for ‘background’ pixels, and Bayes theorem is applied to obtain an image of the posterior probability that each pixel is skin. Skin blob detection is performed over an image pyramid by applying a Difference of Gaussians (DOG) operator [14] to the skin probability image at each level. A face region is declared at local maxima in the DOG response with positive response above threshold, and corresponding high skin probability. The approximate scale of the face is obtained from the pyramid level. Figure 1 shows an example image, skin probability, and detected candidate face regions.

## 2.2 Pose Based Face Rendering

We require a method of rendering faces at poses different from those in the training material. The approach used here is to combine coarse 3-D geometry with multiple texture maps. The model has two parts: a global 3-D geometric model of the head, and a set of visual ‘aspects’ which define appearance over local regions of pose space. The shape of the head is modelled simply as an ellipsoid, the parameters of which are fitted to a single training image of the person. Figure 2a shows a training image for the ‘Basil’ model, and Figure 2b the ellipsoid model overlaid. The aims of using a 3-D model for the head are two fold:



**Fig. 2.** Ellipsoid head model. The triangulation shown (a) is coarser than that used, to aid visibility. The blank area of the texture map is the back of the head, which has not yet been observed.

*1. Extrapolation:* The 3-D model allows us to extrapolate some way from a single view of the person and propose how the person looks in nearby poses. The single training image is back-projected onto the ellipsoid to give a texture map (Figure 2c), then a new view of the head in a different pose can then be rendered by transforming the ellipsoid and projecting the texture map back into the image. Figure 2d shows an example: for poses near to the one from which the texture map was obtained, fairly accurate images can be rendered. Because the ellipsoid geometry only approximates the head shape, the realism of the rendered views degrades as the pose change increases, principally because the ellipsoid does not predict self occlusions (such as the eye being occluded as the face looks down). However, it will be seen that combining a simple shape model with *multiple* texture maps enables accurate rendering of many poses. By contrast, an accurate 3-D model could extrapolate further from a single view, but it is difficult to obtain such an accurate model, and an inaccurate but non-smooth model can introduce many artifacts that we wish to avoid. Ellipsoids [1] and close relatives (superquadrics [11], tapered ellipsoids [13]) have been applied successfully to head tracking by several authors.

*2. Pose space:* The second reason for the 3-D model is that it provides a global reference frame against which any image of the face can be aligned. Initially, having seen just a single image of the face, we have a good idea of the appearance in only a narrow range of poses, and with fixed facial expression. Estimating the pose of a new image and verifying the identity of the person allows a new image to be classified as: (i) close in pose and appearance to an already seen image, (ii) in a pose far from one observed up to this point, or (iii) in a known pose but with differing appearance (facial expression). In the latter two cases the algorithm considers expanding the model by adding additional texture maps, positioning them appropriately in pose space. This allows the model to be improved without manual supervision.

### 2.3 Pose Estimation

Given a candidate face region in the image, the pose of the face is recovered by search in the joint pose/appearance space, proposing the appearance of the face and comparing against the target image. The pose is parameterized as a 6-D vector  $\mathbf{p} = \langle \theta, \phi, \psi, \sigma, \tau_x, \tau_y \rangle$  corresponding to rotation, scale, and 2-D translation





**Fig. 3.** Pose estimation (best viewed in colour). Top rows show original image, middle rows show ellipsoid overlaid at the estimated pose, bottom rows show overlaid model rendered at the estimated pose.

in the image. Rotation is specified by azimuth  $\theta$ , elevation  $\phi$ , and in-plane rotation  $\psi$ . This parameterization allows reasonable bounds to be specified easily. A candidate face region provides an initial estimate of scale  $\tilde{\sigma}$ , up to the scale step between pyramid levels, and translation  $\langle \tilde{\tau}_x, \tilde{\tau}_y \rangle$  (the centre of the candidate region). The task is to find the pose parameters  $\hat{\mathbf{p}}$  which maximize the similarity between the rendered view  $R(\mathbf{p}, \mu)$  and the target image  $I$ . Normalized cross-correlation (NCC), masked by the silhouette of the rendered view, is used as the similarity measure:

$$\hat{\mathbf{p}} = \arg \max_{\mathbf{p}} \left[ \max_{\mu \in \{\mu_{\mathbf{p}}\}} \text{NCC}(I, R(\mathbf{p}, \mu)) \right] \quad (1)$$

For a given pose, *multiple* appearances  $R(\mathbf{p}, \mu)$  are proposed by selecting a subset of the texture maps  $\{\mu_{\mathbf{p}}\}$  which are (i) close to the current pose, and (ii) varying in expression. This is done by first finding the texture map which has pose  $\mathbf{q}$  closest to the current estimate  $\mathbf{p}$ , then selecting all texture maps with pose close to  $\mathbf{q}$  (which represent different facial expressions). Distance between



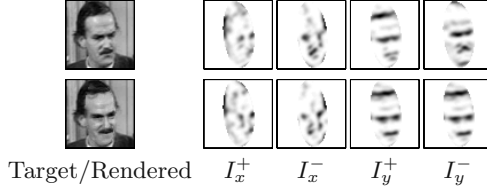
**Fig. 4.** Face classification based on multiple appearance proposals. The leftmost image is the target, with rows showing proposals rendered from the Manuel, Basil and Sybil models. The task is to decide which proposal to accept, or to reject all.

poses is computed by the dot product between a front-facing vector normal to the ellipsoid, so that in-plane rotation about the frontal view does not influence the distance. Using this ‘nearest neighbour plus siblings’ approach to selecting texture maps allows the algorithm to consider texture maps corresponding both to close poses and varying facial expression. Numerical optimization is carried out using the coordinate descent algorithm of [8]. Figure 3 shows examples of pose estimation. Additional examples can be seen in Figure 4, discussed below.

## 2.4 Classification

Given an estimated pose, a set of images is proposed by the models of each person. Figure 4 shows an example, with each person model attempting to reproduce the leftmost image, of Basil. Note that the proposals here have the same pose but vary in facial expression. The aim now is to obtain a representation of the face image suitable for person classification, capturing the essential structure of the facial appearance but allowing for small local misalignments between the original and rendered images due to factors such as the approximation of the face shape as ellipsoidal. Using this representation, one of the proposed images may be accepted as a match, yielding classification of the person, or all may be discounted, in the case of a non-face region, or person other than those modelled.

Use of ‘edges’ rather than raw grey levels for emphasizing salient image structure has been proposed in many contexts [14] and an edge-based descriptor is used here, proposed most recently for comparing optical flow fields [6]. For an image  $I$ , the image gradients  $I_x, I_y$  are computed, and half-wave rectified to form four non-negative channels  $I_x^+, I_x^-, I_y^+, I_y^-$ . Each channel is then blurred with a Gaussian to give some robustness to local image deformations, and the descriptor for the image  $D(I)$  is formed by normalizing and concatenating the four channels. The non-negativity and relative sparseness of signal in each channel allows the channels to be blurred without destroying orientation information or



**Fig. 5.** Gradient descriptor for target (top) and rendered (bottom) images. Darker grey levels represent larger values. The similarity measure here is 0.98, a close match.



**Fig. 6.** Model update by tracking. A colour tracker successfully tracks the face over large pose variation and is used to validate proposed updates to the model.

edges by cancelling positive and negative gradients. The width of the Gaussian is set proportional to the scale of the face in the image.

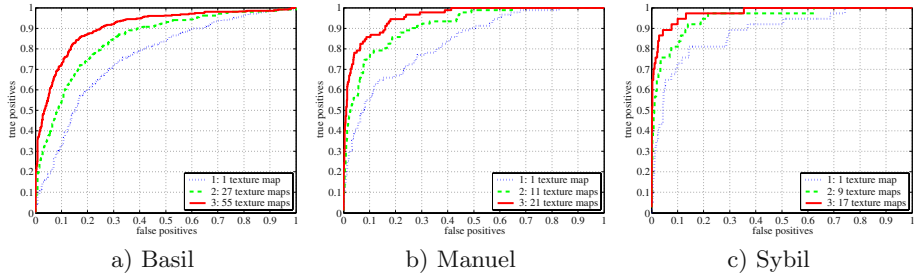
When comparing descriptors for a target image  $I$  and a rendered view of the ellipsoid  $R(\mathbf{p}, \mu)$ , the rendered view is overlaid on the target image (in the manner of Figure 4) before computing image gradients in order to avoid introducing spurious edges due to the ellipsoid boundary. Similarity between the corresponding descriptors  $D(I)$  and  $D(R(\mathbf{p}, \mu))$  is obtained by correlation, considering only pixels within the ellipsoid mask. Figure 5 shows example descriptors for target and rendered images.

## 2.5 Model Learning

The supervision required for learning the face model is minimal: a face for each character is identified in one frame, and the ellipsoid model fitted. Additional training is automatic, as will now be described.

Having computed the similarity (section 2.4) between a set of face candidates and a particular person, a decision is made as to which detections to add to the model as new texture maps, enabling the model to cope with wider variations in expression and pose.

A low threshold on similarity  $t_l$  is defined, above which we are confident that a detection matches a particular person. Three cases then follow: (i) if the similarity of a match is above a second higher threshold ( $t > t_h > t_l$ ) and the pose is close to one already seen, then the image need not be added to the model. (ii) If however the match is certain ( $t > t_h$ ) and the pose is far from one already seen, the image is added to the model so that the range of pose covered is expanded. Finally (iii), less certain matches ( $t_l < t < t_h$ ) which lie close to an existing pose are validated by tracking. These would typically represent unseen facial expressions. To validate such matches, temporal coherence of the video is exploited: a tracker is run from frames with certain matches, ending at the



**Fig. 7.** ROC curves for three characters in 1,500 key frames. Successful identification requires correct detection, pose estimation, and recognition. In all cases, unsupervised model update improves the accuracy of the model.

candidate frame. The tracker used is a colour version of a deformable region tracker [4]. If the position of the tracked region agrees with the detected face, then the model is updated. Figure 6 shows an example of successful tracking over wide pose variation.

### 3 Experimental Results

The algorithm was tested on 1,500 key-frames taken one per second from the episode ‘A Touch of Class’ of the sitcom ‘Fawlty Towers’. We evaluated detection of three of the main characters: Basil, Sybil and Manuel. The task was to detect the frames containing each character, and identify the image position and pose of the face correctly. Correctness was measured by the distance to ground truth points marked on the eyes, nose and ears according to pose, requiring distance of all predicted points to be less than 0.3 of the inter-ocular distance. Corresponding points for the model (for testing purposes only) were obtained by back-projecting the ground truth points onto the ellipsoid during training and model update. Pose of the ground truth faces in the video covers poses of around  $\pm 60^\circ$  azimuth,  $\pm 30^\circ$  elevation and  $\pm 45^\circ$  in-plane rotation. Faces vary in scale from 15 to 200 pixels. The values of the thresholds  $t_l$  and  $t_h$  were determined from a validation set and kept fixed throughout the experiments.

Figure 7 shows ROC curves for each of the three characters. Note that we treat the problem as one of detection rather than 1-of- $m$  classification since we do not know *a priori* all the characters in the video. For each character, curves are shown for the initial model and two runs of the model update procedure. The number of texture maps after model update varied for each character, due to the varying number of frames in which the character appears and differences in pose variation between characters, and is shown in the legend. The graphs show clear improvement in the accuracy of the model after update, for example in the Basil model the equal error rate decreases from 30% to 15% after two rounds of update. At this stage, characters can be detected in 75–95% of frames at a false positive rate of 10%. These results are extremely promising given the difficulty of the task. It is interesting to observe that the performance on Sybil

is notably better than the other characters; this is the ‘moustache problem’ - the moustache is a strong visual feature shared between Basil and Manuel, and indeed three other secondary characters in the episode, which gives much scope for confusion.

## 4 Discussion

We have presented methods for detecting and identifying characters in video across wide variations in pose and appearance by combining a simple 3-D model with view-dependent texture mapping. Placing the views of the face in a common reference frame allows more efficient search than possible with an unorganized collection of images, and provides a basis for automatic model update. Use of a simple 3-D model rather than a detailed face model [3] avoids introducing severe rendering artifacts due to incorrect modelling of self-occlusion, and multiple texture maps allow facial expressions to be modelled, which is challenging for 3-D models with a fixed texture map.

**Acknowledgements.** Thanks to EC Project CogViSys for funding.

## References

- [1] S. Basu, I. Essa, and A. Pentland. Motion regularization for model-based head tracking. In *Proc. ICPR*, pages 611–616, 1996.
- [2] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE PAMI*, 19(7):711–720, 1997.
- [3] V. Blanz, S. Romdhani, and T. Vetter. Face identification across different poses and illumination with a 3D morphable model. In *Proc. AFGR*, 2002.
- [4] Y. Chen, T. Huang, and Y. Rui. Optimal radial contour tracking by dynamic programming. In *Proc. ICIP*, 2001.
- [5] T. F. Cootes, K. Walker, and C. J. Taylor. View-based active appearance models. In *Proc. AFGR*, pages 227–232, 2000.
- [6] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *Proc. ICCV*, 2003.
- [7] S. Eickeler, F. Wallhoff, U. Iurgel, and G. Rigoll. Content-Based Indexing of Images and Video Using Face Detection and Recognition Methods. In *Proc. ICASSP*, 2001.
- [8] V. Ferrari, T. Tuytelaars, and L. Van Gool. Wide-baseline multiple-view correspondences. In *Proc. CVPR*, pages 718–725, 2003.
- [9] A. W. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. In *Proc. ECCV*, volume 3, pages 304–320. Springer-Verlag, 2002.
- [10] B. Heisele, T. Serre, M. Pontil, and T. Poggio. Component-based face detection. In *Proc. CVPR*, pages 657–662, 2001.
- [11] N. Krahnstoever and R. Sharma. Appearance management and cue fusion for 3D model-based tracking. In *Proc. CVPR*, pages 249–254, June 2003.
- [12] S. Z Li, L. Zhu, Z. Q. Zhang, A. Blake, H. J. Zhang, and H. Shum. Statistical learning of multi-view face detection. In *Proc. ECCV*, 2002.

- [13] M. C. Lincoln and A. F. Clark. Pose-independent face identification from video sequences. In *Proc. BMVC.*, 2001.
- [14] D. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, pages 1150–1157, 1999.
- [15] H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied to faces and cars. In *Proc. CVPR*, 2000.
- [16] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. CVPR*, pages 511–518, 2001.
- [17] W. Zhao, R. Challa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35:399–458, 2003.

# Content Based Image Synthesis

Nicholas Diakopoulos, Irfan Essa, and Ramesh Jain

GVU Center / Georgia Institute of Technology

{nad|irfan|rjain}@cc.gatech.edu

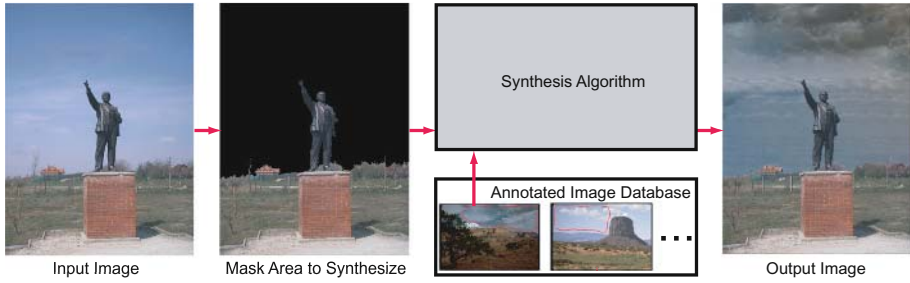
**Abstract.** A new method allowing for semantically guided image editing and synthesis is introduced. The editing process is made considerably easier and more powerful with our content-aware tool. We construct a database of image regions annotated with a carefully chosen vocabulary and utilize recent advances in texture synthesis algorithms to generate new and unique image regions from this database of material. These new regions are then seamlessly composited into a user's existing photograph. The goal is to empower the end user with the ability to edit existing photographs and synthesize new ones on a high semantic level. Plausible results are generated using a small prototype database and showcase some of the editing possibilities that such a system affords.

## 1 Introduction and Related Work

The ongoing digital media revolution has resulted in an untold amount of new digital content in the form of images and videos. Much of this digital content is generated by capturing real scenarios using cameras. We feel that digital media production would become much simpler, more effective, and cheaper if intuitive tools were available for combining *existing* photos to generate *new* images. Merging images and image segments together to form new images has been around for a long time in the form of image and video compositing. The composition process is however tedious and best carried out by the practiced eye of an artist. For such synthesis an artist combines various image regions from different sources to achieve a predefined content and context in the final image. We seek to simplify this process by providing a content-aware tool for generating new images.

To facilitate this form of semantic image synthesis, we first create a *database* of imagery which has regions annotated with semantic labels and image characteristics. Then, based on the content that the user desires, the user can *query* this database for imagery that will suit the region that will be composited. The user then chooses an image region from the query results, which acts as a source for texture *synthesis* algorithms [1,2,3]. The synthesized region is composited into the image being edited. The system pipeline is shown in Fig. 1. In this way a user can edit a photo by synthesizing any number of user defined regions from an existing database of imagery. We have termed this semantically guided media recombination process Content Based Image Synthesis (CBIS).

At the heart of the CBIS method is a reliance on semantic annotations of regions (e.g. sky, mountain, trees etc.) so that if the user wants to synthesis a new mountain range into his photo he can search the database for other mountain regions. The idea of using such



**Fig. 1.** The flow of the CBIS application. An input image is loaded and the area to replace masked. A query against an annotated image region database is then made by a user in order to find suitable content with which to fill the area. This source region is then used by a texture synthesis algorithm to produce a new region which is finally composited into the input image.

high level annotations for doing search and synthesis has been applied successfully by Arikan [4] in the domain of motion data for animation. Their vocabulary, consisting of such verbs as *walk*, *run*, and *jump* is used to annotated a database of motion data. When the vocabulary is applied to a timeline, the system synthesizes plausible motion which corresponds to the annotated timeline.

The analogue in the image domain is the segmentation and annotation of distinct image regions with high level semantic tags. Zalesny [5] segments textures into subtextures based on clique partitioning. Each subtexture is represented by a label in a spatial label map. The label map guides their texture synthesis procedure such that the correct sub texture appears in the correct spatial position. Hertzmann [6] also defines some notion of a label map in his texture-by-numbers synthesis scenario. Our method borrows from these ideas for a label map but defines a much higher level mapping, less based on image statistics and more based on the semantic annotation of the region. For example, the user may desire one region to be synthesized as *sky* and another region as *mountain*. These region annotations ultimately map to the regions from the database that have been selected by the user as synthesis sources.

In the next section we further motivate the semantic power of such a CBIS application using semiotics. In section 3 we detail our methods for segmentation and annotation of images. Section 4 lays out our query method and details the hybrid texture synthesis approach used. Finally, we show some results that have been generated with the system in 5 and conclude with a number of directions for future work.

## 2 Semiotics Basics

*Denotation and Connotation:* Semiotics provides us with a rich set of abstractions and quasi-theoretical foundations supporting our recombinant media application. A user may want to composite a newly synthesized region into an existing photograph for practical purposes, aesthetic reasons or, from a semiotic perspective, to change the *meaning* of the photograph. Barthes [7] identifies several ways in which one can alter the connotation of an image (e.g composition, pose, object substitution/insertion, photogenia etc.). Our



system thus uses image composition to enable the user to alter both the denotation (pixels) and the connotation. For example, given the photo of the stalin statue with the happy sky seen in Fig. 1 (input), we might want to substitute a cloudy or stormy sky (output). The resultant change in meaning is left to the subjective interpretation of the reader.

*Structural Analysis:* Semiotic systems can be defined structurally along two primary axes: *syntagm* and *paradigm*. The syntagm defines the spatial positioning of elements in an image, whereas the paradigm is defined by the class of substitutions that can be made for a given element [7,8,9]. A linguistic analogy is usually helpful for understanding. The syntagm of a sentence corresponds to its grammatical structure (e.g. noun-verb-prepositional phrase); the paradigm corresponds to the set of valid substitutions for each word in the sentence. For instance, a verb should be substituted with a verb in order that the sentence still make sense.

Syntagm and paradigm are the structural axes along which a user may vary an image in the course of editing. We can consider three combinations of variation along these axes: (1) vary the paradigm and fix the syntagm; (2) fix the paradigm and vary the syntagm; or (3) vary both the paradigm and syntagm. (1) is analogous to playing with a Mr. Potatoe Head: the structure of the face remains the same, but various noses, mouths, etc. can be substituted into that structure. (2) roughly corresponds to the method of texture-by-numbers in [6]. In that work, a new syntagm (a label map) is drawn by a user and filled in with the corresponding labelled regions from a source image. (3) defines a more difficult operation since changing the syntagm of an image can change the valid paradigmatic substitutions as well. We consider variation (1) in this work. The layout of the input photo remains the same (i.e. mountains stay mountains of the same shape), but the database provides a set of paradigmatic variations (i.e. rocky mountains can become snowy mountains).

### 3 Database Generation

The database creation process consists primarily of segmenting meaningful and useful regions in images and annotating them with the appropriate words from our annotation vocabulary. Each of these segmented, annotated regions is then stored in an XML structure which can later be queried by the end user. In our prototype system we have annotated slightly more than 100 image regions which serve as the database.

*Region Segmentation:* The first step in generating the database of images is the segmentation of meaningful regions in these images. While there has been some recent progress in the automatic segmentation of semantically meaningful regions of images [10,11,12], or even semi-automatic segmentation, currently we opt for a fully manual procedure as this allows for a more directed user input of higher semantic import.

*Region Annotation:* The annotation vocabulary is chosen to fit the domain of natural landscape imagery. This domain makes sense for us since textures are prevalent and because a relatively small vocabulary can describe the typical regions in such a scene.

The choice of words used to annotate quantities like hue, saturation, and lightness is informed by [13,14]. The vocabulary follows:

Region: {Sky | Mountains | Trees | Vegetation | Water | Earth | Hue | Lightness | Saturation | Distance}  
 Sky: {Clear | Partly Cloudy | Mostly Cloudy | Cloudy | Sunset | Sunrise | Stormy}  
 Mountains: {Snowy | Desert | Rocky | Forested}  
 Trees: {Deciduous | Coniferous | Bare}  
 Vegetation: {Grass | Brush | Flowering}  
 Water: {Reflective | Calm | Rough | Ocean | Lake | Stream | River}  
 Earth: {Rocky | Sandy | Dirt}  
 Hue: {Red | Orange | Yellow | Green | Blue | Purple | Pink | Brown | Beige | Olive | Black | White | Gray}  
 Lightness: {Blackish | Very Dark | Dark | Medium | Light | Very Light | Whitish}  
 Saturation: {Grayish | Moderate | Strong | Vivid}  
 Distance: {Very Close | Close | Medium | Distant | Very Distant | Changing}

Region annotations are made manually using a GUI. Relying on a fully manual process allows us to work with higher level semantic categories. Of course, as automatic annotation methods get better, they can be integrated to supplement the manual procedure. Given the categories chosen above, there should be little problem with consensus on the appropriate annotation(s) for a given region, though in general subjectivity can be a problem for manual annotation.

The hue, saturation, and lightness user annotations are augmented by fuzzy histograms of HSL pixel values. A 13 bin hue histogram, a 4 bin saturation histogram, and a 7 bin lightness histogram are generated based on the HSL pixel values in a given region. Bins are fuzzy insofar as a given pixel can contribute (bi-linearly) to adjacent bins. Each bin also corresponds to one of the vocabulary words for that category; for saturation the 4 bins correspond to <Grayish, Moderate, Strong, Vivid>. A saturation histogram of <.8, .2, 0, 0> therefore indicates a very grayish region.

This dual representation of lower level features should also mitigate the somewhat subjective user annotations. Currently, we are also studying other automatic or semi-automatic methods for annotating entities such as the lighting direction, or camera perspective. These would serve to make database query results even more pertinent to the image into which they will be synthesized.

## 4 Image Recombination

*Query:* The image recombination procedure begins with the user defining a region that will be replaced in his input image (e.g. a mountain range is selected). This region is then annotated with keywords from the vocabulary using a drag and drop GUI. This annotation is used to query the XML database and return the  $N$  most pertinent images from which the user selects a source image. Subtle decisions in lighting, perspective, and color are thus not made automatically and can be evaluated by the user. This maximizes the user's potential to affect connotation in the image since he has good suggestions from the database, but ultimately has the final choice in the paradigmatic substitution.

Matching of annotated regions proceeds as suggested by Santini [15]. This approach allows for binary feature sets (i.e. presence/absence of keywords) to be compared with fuzzy predicates. Thus we can compare keyword annotations of a region's lower-level features (e.g. saturation) with the histograms of those features as calculated directly from pixel values.

Let  $\hat{f}(R_i) = \langle f_1(R_i), f_2(R_i), \dots, f_p(R_i) \rangle$  represent a  $p$ -dimensional feature vector for a region  $R_i$ .  $f_k(R_i) = 1$  if region  $R_i$  has the keyword annotation associated with feature  $k \in \{1 \dots p\}$ . For fuzzy features such as saturation we maintain two feature vectors with  $f_k(R_i) \in (0, 1)$ . One vector is based directly on the histogram of pixel values. The other vector is based on the keyword annotation, but is also made fuzzy. As an example let's consider the two feature vectors describing saturation for  $R_1$ . The feature vector calculated from pixel saturation values might be:  $\hat{f}(R_1) = \langle 0, .1, .7, .2 \rangle$ . If  $R_1$  also has the *strong* keyword annotation the second feature vector is  $\hat{f}(R_1) = \langle 0, .25, 1, .25 \rangle$ . This fuzzyness allows for more meaningful retrieval results since it allows for a smoother range of similarity scores. In addition to hue, saturation, and lightness, we make the distance vector fuzzy since it also benefits from a gradient score. In general, any attribute which can naturally be described using an intensity scale benefits from a fuzzy representation.

Based on [15] a symmetric similarity function  $\sigma$  is defined between two feature vectors in the following way,

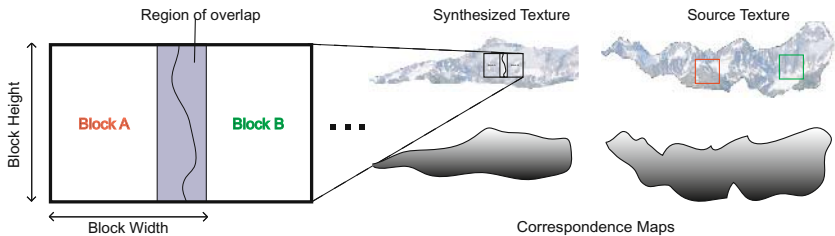
$$\sigma(\hat{f}(R_a), \hat{f}(R_b)) = \sum_{i=1}^p \min(f_i(R_a), f_i(R_b)) \quad (1)$$

The dissimilarity  $\delta$  can be written as,  $\delta(\hat{f}(R_a), \hat{f}(R_b)) = p - \sigma(\hat{f}(R_a), \hat{f}(R_b))$ . We maintain a feature vector for each semantic category of each region, though we could concatenate these vectors and arrive at the same result. Comparing two image regions then consists of computing dissimilarity scores for each semantic category and summing them to arrive at an aggregate dissimilarity score between those two regions. Where necessary the dissimilarity of two regions also takes into account the dual feature vector representation by equally weighting the histogram vector and fuzzy keyword vector in the final score.

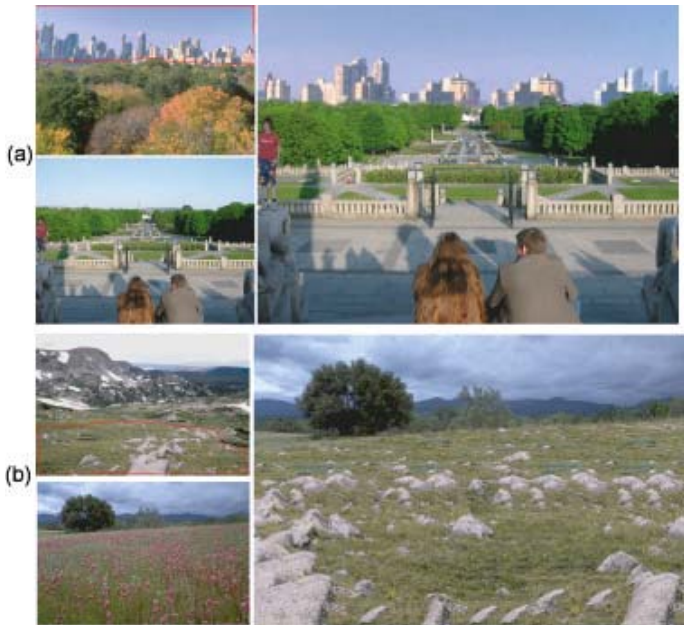
*Synthesis:* In general there are many methods for doing image based synthesis. Here we focus specifically on using patch-based texture synthesis algorithms such as those of [1,2,3]. Patch-based approaches work by copying small regions of pixels from a source texture such that when these regions are stitched together they give the same impression as the original texture. The output texture need not be the same shape or size as the source.

In particular we have implemented the method of texture synthesis and transfer detailed by Efros in [1]. Our pixel blocks are rectangular and tiled over the synthesis plane with overlapping regions through which seams are optimally cut using a dynamic programming algorithm (see Fig. 2). Each successive block of pixels is chosen by scanning the source texture for areas that will minimize a euclidean error metric as measured against imbricated adjacent areas. As many textures in our application are non-stationary and change due to perspective effects, we also use vertical correspondence maps when calculating the error metric. This ensures that parts of a source texture far away (i.e. at the top of a region) are used as samples when synthesizing the parts of the destination texture region that are also far away. Though this seems to work well in practice, the amount of perspective can vary considerable between source and destination. Vertical correspondence maps can fail to generate convincing results in such cases.

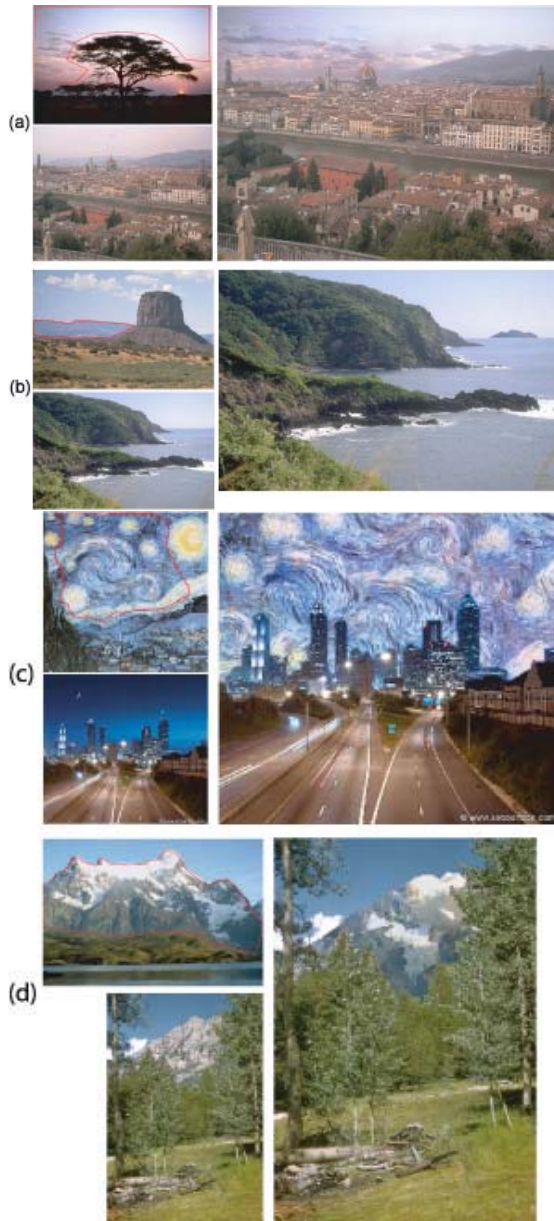
Though the dynamic programming algorithm for seam generation detailed in [1] does a decent job for textures with some high frequency content, it falters for more slowly changing textures such as a sky gradient. In these cases, we smooth these seams with a technique in which a Poisson equation is solved across the boundary of the seam [16]. This has the effect of creating smooth transitions between blocks without sacrificing the saliency of the underlying texture.



**Fig. 2.** A texture is synthesized. The blow up shows two overlapping blocks taken from the source and the seam between them. Vertical correspondence maps guide sampling from the source.



**Fig. 3.** Results generated using the CBIS application. Each block (a,b) consists of a source texture outlined in red (upper left); an input image (lower left); and the output (right).



**Fig. 4.** Results generated using the CBIS application. Each block (a,b,c,d) consists of a source texture outlined in red (upper left); an input image (lower left); and the output (right).

There are a few algorithm parameters that are worth mentioning briefly here. The block size must be chosen large enough to capture the largest feature or texon size of the source texture. Choosing a value too small can lead to synthesized results which lack

the structure of the original. Additionally, we allow the algorithm to iterate across the whole synthesized texture a variable number of times such that the initial pass is done with larger blocks (thus “laying out” the texture) and subsequent passes done with a smaller block size such that finer details are preserved.

## 5 Results

Queries were performed against a small proof of concept database of about 50 images, representing just over 100 distinct image regions. Synthesis timings vary according to parameters and region size in both the source and destination images, but were in the range of about 2 to 30 minutes for a Java implementation on a 2.4 GHz processor.

Some example images generated using our CBIS system are shown in Fig. 3 and Fig. 4. Results can also be viewed online at (<http://cpl.cc.gatech.edu/projects/CBIS/>); we encourage viewing of results digitally. In Fig. 3 (a) a city skyline is inserted; (b) a field of flowers is replaced with a field of rocks. Fig. 4 contains additional results: (a) a gentle sunset replaces a cloudy sky over Florence; (b) a distant island is inserted; (c) the night sky over Atlanta is replaced using the *Starry Night* painting by van Gogh leading to a unique blend of photograph and impressionist painting; (d) a rocky mountain is replaced with a snowy mountain.

The parameters for each of the images in Fig. 3 and Fig. 4 were chosen carefully. In particular the block size was chosen to be slightly larger than any repeatable features in the source. All sky replacements utilized the Poisson smoothing functionality. Vertical correspondence maps were used for all source and destination region pairs with the exception of Fig. 4 (c). Also, two synthesis passes were made for each output image.

## 6 Conclusions and Future Work

We have introduced a method for the content-based generation of new images. A user defines a region in his image to replace and then queries a database of images annotated with region semantics and image characteristics to find a suitable source image. The chosen source image is then used in a texture synthesis step to produce the final output image. Results, such as those seen in Fig. 3 and Fig. 4, are visually convincing.

Invoking semiotic theory allows us to view such a CBIS tool as a powerful way of changing meaning in photographic content through careful substitution and insertion of image elements. The extent of this editing power is dictated only by the size and breadth of the underlying database, and of the vocabulary used to annotate it. Thus, there are several obvious areas of future work such as expanding the database, annotating image regions with additional visual information such as perspective or lighting characteristics, and increasing the effective size of the query vocabulary by tying into a system such as WordNet. To expand the database we would specifically like to explore adding segmented objects to the database. A substitute object could then have the area around it filled in using a hole filling algorithm such as [17]. Improvements also need to be made in the synthesis phase, such as accounting for lighting effects (e.g. shadow in Fig. 1) or interactions between adjacent textures in the final image.

Another important direction for future work is in applying a content-based synthesis framework to other types of media such as video or audio. To be successfully applied in each of these domains several things must first be defined: (1) appropriate segmentation methods and units, (2) annotation vocabulary and low-level features, and (3) synthesis algorithms that combine the segmented units so that the output is believable. In short, workable segmentation and integration algorithms must be found for these other types of media.

**Acknowledgements.** Thanks to Derik Pack for his help implementing elements of the XML storage system. We appreciate the helpful comments of Stephanie Brubaker in preparing this paper. We also acknowledge the copyright holders of image segments used, many of which were downloaded from <http://elib.cs.berkeley.edu/photos> or elsewhere on the internet.

## References

1. Efros, A.A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: Proceedings of ACM SIGGRAPH. (2001) 341–346
2. Ashikhmin, M.: Synthesizing natural textures. In: Symposium on Interactive 3D Graphics. (2001) 217–226
3. Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A.: Graphcut textures: Image and video synthesis using graph cuts. Proceedings of ACM SIGGRAPH (2003) 277–286
4. Arikan, O., Forsyth, D.A., O'Brien, J.F.: Motion synthesis from annotations. In: Proceedings of ACM SIGGRAPH. (2003) 402–408
5. Zalesny, A., Ferrari, V., Caenen, G., VanGool, L.: Parallel composite texture synthesis. In: ECCV Texture Workshop. (2002) 151–155
6. Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. In: Proceedings of ACM SIGGRAPH. (2001) 327–340
7. Barthes, R.: Image, Music, Text. The Noonday Press, New York (1977)
8. Chandler, D.: Semiotics: The Basics. Routledge, New York (2002)
9. Manovich, L.: The Language of New Media. MIT Press, Cambridge, MA (2001)
10. Duygulu, P., Barnard, K., de Freitas, J.F.G., Forsyth, D.A.: Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In: Proceedings ECCV, Springer-Verlag (2002) IV: 97–112
11. Malik, J., Belongie, S., Shi, J., Leung, T.K.: Textons, contours and regions: Cue integration in image segmentation. In: Proceedings of ICCV. (1999) II: 918–925
12. Singhal, A., Luo, J., Zhu, W.: Probabilistic spatial context models for scene content understanding. In: Proceedings of CVPR. (2003) 235–241
13. Mojsilovic, A.: A method for color naming and description of color composition in images. In: Proc. Int. Conf. Image Processing (ICIP). (2002) 789–792
14. Corridoni, J.M., Del Bimbo, A., Pala, P.: Image retrieval by color semantics. *Multimedia Syst.* **7** (1999) 175–183
15. Santini, S., Jain, R.: Similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **21** (1999) 871–883
16. Perez, P., Gangnet, M., Blake, A.: Poisson image editing. In: Proceedings of ACM SIGGRAPH. (2003) 313–318
17. Criminisi, A., Perez, P., Toyama, K.: Object removal by exemplar-based inpainting. In: Proceedings of CVPR. (2003) II: 721–728

# Interactive Content-Based Retrieval Using Pre-computed Object-Object Similarities

Liudmila Boldareva and Djoerd Hiemstra

University of Twente, Databases Group,  
P.O.Box 217, 7500 AE Enschede, The Netherlands  
{L.Boldareva,D.Hiemstra}@utwente.nl

**Abstract.** We propose using truncated object-object similarity matrix as an access structure for interactive video retrieval. The proposed approach offers a scalable solution to retrieval and allows combination of different feature spaces or sources of information. Experiments were performed on TREC Video collections of 2002 and 2003.

## 1 Introduction

With rapid development of digital media in the past decade, *Content-based information retrieval* (CBIR) has become an active research area. Originally meant for text documents, information retrieval quickly became dearly needed for other media such as still images and video. Though CBIR usually suggests the retrieval of non-textual information, the term does not exclude text documents<sup>1</sup>. The goal of a retrieval system is to satisfy the *information need* of the user. The information need is communicated to the system, e.g. by providing an example query.

A number of approaches to CBIR exist. The pioneering image retrieval systems used large experience existing in the text retrieval domain, successfully adopting the vector space model [7,14,10]. Probabilistic approaches from text retrieval (e.g. [9,12] gained less popularity among non-text CBIR researches with some notable exceptions [3,19,16]. One of the reasons lies in the difficulty of translating the lower-level features into probability values. Other recent research is inspired by machine learning methods. Self-organising maps [11] and support vector machines [4,15] are employed to solve the problems of CBIR. Many existing retrieval systems rely on active participation of the searcher in the retrieval process, which is known as *relevance feedback* [13].

Regardless of the approach used, a retrieval system should be able to ‘understand’ the users’ information need and provide him/her with satisfactory answers. The problem is that high-level content of a document, in the way a human being understands it, is hard to translate into a machine-language concept with current techniques for automatic lower-level feature extraction. Rich feature spaces might be created in an attempt achieve a correspondence between lower-level features and human perception. This immediately creates a

---

<sup>1</sup> In this paper the term ‘document’ is used in a broad sense, implying any source of information such as text, images, videos, etc.



disadvantage—a high-dimensional space that is not well suited for fast access via indexing. It raises the scalability problem: methods that perform well on small collections can not be used on a collection of usable size, due to the ‘dimensionality curse’ [6]. In this paper we propose a framework for content-based indexing and retrieval, that

- is able to use any available technique for feature extraction, and allows easy combination of different sources of information;
- focuses on relevance feedback as an important component of the information retrieval process;
- allows efficient interaction with the user, i.e. it offers a solution to the scalability problem.

We present a description of the proposed framework in Sec. 2. Interaction between the system and the user is studied in Sec. 3. Experiments performed on the collection of the TREC Video retrieval workshop (TRECVID) [1] are presented in Sec. 4. Conclusions and future work directions can be found in the last section.

## 2 Probabilistic Indexing and Retrieval

Consider a collection  $\mathcal{I}$  of information objects  $i$  among which there is one that the user is looking for, the *search target* denoted  $T$ <sup>2</sup>. During the search process, the system presents the user with intermediary retrieval results. The user can indicate which examples are *relevant* to his/her information need, those are *positive examples*. If an object is *not relevant* to the query, the user may indicate so, thus providing the system with *negative examples*. Given the feedback information, the retrieval system produces a new set of candidate documents to be assessed by the user. There may be several loops of *relevance feedback* during one search session.

We want to make use of the notions ‘relevant’ and ‘non-relevant’ without having to refer to lower-level (image) features. We do so by relating objects in the collection to each other. A binary variable  $\delta_i$ , that takes values 1 and 0, denotes the events of *positive* and *negative feedback* respectively. For two documents the following reflects their ‘measure of closeness’:  $P(\delta_i = 1|T)$ , the probability of an object  $i$  marked by the user as relevant given that  $T$  can be referred to as the target for the search. When unambiguous, we use a shorthand notation  $P(\delta_i|T)$ .

### 2.1 Interactive Retrieval in a Probabilistic Framework

For interactive retrieval we use a probabilistic approach. The idea is to predict the set of documents relevant to the user’s information need, based on his/her request, accompanied by feedback, and the data representation (i.e. our measure of closeness  $P(\delta_i|T)$ ). Using Bayes’ rule the problem can be stated as estimating the probability of relevance  $P(T)$  given user’s feedback  $\delta^1, \dots, \delta^n$  and the collection indexing [12,3,16].

<sup>2</sup> The search target may be a single document, but it can as well be a number of documents covering a certain subject satisfying the user’s information need.

We write it down in the following iterative form, using the assumption that the  $\delta^1, \dots, \delta^n$  are conditionally independent given the target  $T$  :

$$P^{\text{new}}(T) = P(T|\delta^1, \dots, \delta^n) = \frac{P^{\text{old}}(T) \prod_{s=1}^n P(\delta^s|T)}{P(\delta^1, \dots, \delta^n)} . \quad (1)$$

We distinguish the following factors that influence an interactive search session:

1. The input provided by the user who is assumed to be reasonable in his/her query formulation and feedback.
2. The current document representation. Within one search session, the indexing of the collection is a static component of the model.
3. The prior information about the relevance of documents in the collection.

Below we describe our approach to indexing of a multimedia collection.

## 2.2 Indexing: The Structure of the Association Matrix

Documents in the collection and their conditional probabilities  $P(\delta_i|T)$  can be visualised as a directed graph with objects  $i \in \mathcal{I}$  as nodes and arcs with weights  $P(\delta_i|T)$  connecting them. In this way each object is *described* by its *associations* with a number of other objects linked to it. We call such representation of the collection an *association matrix*, denoted  $\mathbf{M}$ .

Ideally we want the associations to refer to high-level semantics (e.g. coming from users' judgements) which might not be achieved using lower-level features. Starting at the point when we do not have knowledge about the human perception of similarity, the associations need to be based on something different. We propose to bootstrap the process by basing the associations on a similarity measure on lower-level features, such as colour, texture, or shapes present in an image (e.g. as used in [7,16]). Typically such similarity measures take values in  $\mathbb{R}$  or  $\mathbb{R}^+$  and thus cannot be directly used as an initial estimate for  $P(\delta_i|T)$ .

In our model we take pair wise similarities based on, e.g., pictorial features, and we are looking for an appropriate transformation to obtain probabilities. Any increasing function with the domain  $\mathbb{R}$  and the range  $[0, 1]$  could suit. When deciding the probabilities in our model, we would like to achieve equal emphasis of the alike similarities and obtain probabilities, uniformly distributed in  $[0, 1]$ . The used transformation spreads the observations evenly on this interval according to their probability of occurrence and not the magnitude of the similarity measure. As a result it reduces the influence of outliers and preserves the scale of the similarities between documents and 'improves the discrimination capabilities of the similarity measures' [2]. Since a priori we cannot prefer some documents of the collection to others in the sense of the distribution of  $P(\delta_i|T)$ , the underlying similarities are assumed to be random values conform to the same probability distribution—the normal distribution.

We transform the computed similarities by subtracting the sample mean and dividing by the sample standard deviation and then applying the standard normal cumulative distribution function, to obtain estimates of the probabilities

which are denoted by  $P(\delta_i|T)$ . The value of  $P(\delta_i = 0|T) = 1 - P(\delta_i = 1|T)$  obtained in this way can be interpreted as a *P-value*, the probability that a variable assumes a value greater than or equal to the observed one strictly by chance [18]. Thus by specifying some  $\alpha$  such that  $P \leq \alpha$  only *significant* pairwise similarities and their corresponding  $P(\delta_i|T)$  are taken into account, and the rest is replaced by an appropriate constant further denoted by  $\bar{p}$ . When updating  $P(T)$  for each object in (1),  $P(\delta_i|T)$  is substituted with  $\bar{p}$  if it is below  $1 - \alpha$ . Here  $1 - \alpha$  serves as a cut-off threshold for the right tail of the distribution. For the rest of the paper the corresponding threshold for the left tail is set to zero. A pair of documents  $i, T$  having their  $P(\delta_i|T)$  significant, are called *neighbours*.

Keeping only neighbours for each element makes the association matrix sparse, which allows faster access to the data. In our experiments with an appropriate/optimal choice of the cut-off threshold, depending in particular upon the size of the collection, the association matrix can grow as slowly as linear without the loss of the search quality. Pre-computed probabilities allow easy combination of different modalities of otherwise hard to combine feature spaces, such as visual information from a shot and speech transcripts from spoken words [8].

### 3 Modelling Interaction for Retrieval

**The user feedback.** During the search session, the current probability of an element to be the user's search target  $P(T)$  is updated according to (1). Every document can be either relevant to the user's information need or not, i.e. the events are disjoint:  $P(\delta_i = 1|T) = 1 - P(\delta_i = 0|T)$ . The objects that are not marked by the user as relevant, take part in the probability update as if they are explicitly rejected by the user.

To ensure that in the lack of positive examples, the excessive (implied) negative feedback does not bury the precious positive examples, the  $\bar{p}$  is set in our experiments to a value in the interval  $(0, \alpha)$ , with the effect that in the ranked list of results the non-neighbours of negative examples do not precede the neighbours of known (if any) positive examples from the last iterations.

**New display for the next iteration.** Upon updating  $P(T)$ , a new set of objects should be presented for relevance judgement, to receive new evidence from the user. The display update is an important part of the search process, since efficiency and quality of retrieval depend on it. Each iteration should bring the user closer to his/her target object. 'Closer to the target' may have various interpretations, such as: the posterior probability  $P(T)$  of the desired information object(s) tends to 1; or the target object approaches the top of the ranked list, etc. The goal of the search is not only to satisfy the users' need, but to do it in few iterations and/or in a limited amount of time. In this paper we report experiments performed with the following display update strategies.

*Best-target strategy.* Following *probability ranking principle* [12],  $P(T)$  is treated as a score that the element receives during retrieval session. The next display set consists of (new) documents that have largest values of  $P(T)$ .

The Best-target strategy is plausible for the user unfamiliar with content-based retrieval (thus, the majority of potential users). The screen always contains objects that are the neighbours of good examples marked by the user. The user is able to observe the immediate result of his/her action. It is not clear however, whether this approach converges the search to the target quickly enough. Cox et al. [3] report that the Best-target search occasionally gets stuck in an isolated ‘island’ of non-relevant documents that are similar to each other only.

*Non-deterministic strategies.* The Randomised display set consists of objects picked from the collection at random. Uniform sampling may give relatively good representation of the collection, which supposedly allows to find the relevant documents quickly. Sampling could be especially useful at the beginning of a search session, when the system has little knowledge about the user information need. When, after a number of iterations, the mass  $P(T)$  is concentrated on a small (relevant) subset of the collection, sampling the *whole* data becomes useless and may have negative effect on the search quality. To minimise this effect, Random-of-Best strategy makes the selection among those objects for which their probability to be the target increased since the last iteration, which are effectively neighbours of relevant examples, and/or not-neighbours of the non-relevant ones. Ideally, the number of elements of which  $P(T)$  increases should shrink on to the group of documents that satisfy the user’s information need.

## 4 Experiments and Evaluation

### 4.1 Interactive Experiment Setup

We use video data provided in the framework of TRECVID. The videos are segmented into shots, and from each shot a representative key frame is extracted. Conditional probabilities for the association matrix are estimated using a generative probabilistic retrieval model (see for detail [19]):

1.  $M^V$  using on Kullback-Leibler divergence as similarity measure for Gaussian Mixture Models built on pictorial data;
2.  $M^t$ , using language model-based similarity on text from speech transcripts;
3.  $M^{vt}$ , a run-time combination of the two modalities, which adds up the relevance scores achieved in both matrices.

We conducted an empirical study on performance difference caused by the prior distribution of the probability of relevance. In order to provide a better than uniform prior probability of relevance, for a number of experiments the text from search topic descriptions serves as a query to match against the speech transcripts, using a language model [9]. In another version of the system the prior distribution is determined by the number of neighbours in the association matrix for each document, so that a document with many neighbours has higher chance to be displayed. This is useful when no prior information about the information need is available, for instance in un-annotated data, or when the query terms typed by the user, do not occur in the collection.

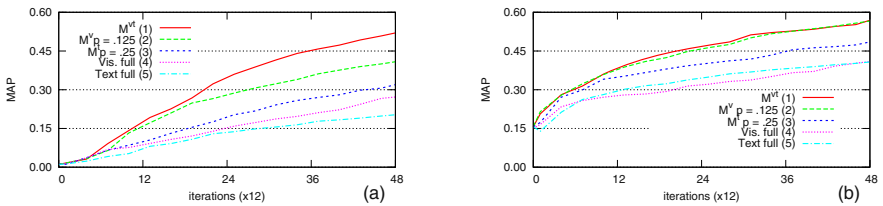
A retrieval session starts with browsing a display set of 12 key frames generated by the prior distribution of  $P(T)$ , which might be based on an initial text query. The user does not have to provide an example query image.

The documents in the ranked list are ordered by the decreasing probability of relevance. A standard TREC evaluation metric, mean average precision (MAP) is used as a measure of user's satisfaction (see [5, Appendix]). Where questionable, signed rank test is used to determine if a difference in performance between two methods is significant. If not stated otherwise, the significance level is  $p \leq 0.05$

## 4.2 Automated Experiments

In the series of experiments, referred to as *automated* the user input has been replaced with relevance judgements available from TREC assessors who played the role of a 'generic user'. The experiments have been performed on a subset of the collection selected so that that half of the key frames was relevant to at least one of the 25 topics. The goal of such setup was to test the retrieval performance in our probabilistic framework, and to find optimum settings to be used in the experiments with real users.

**Values in the association matrix.** Values of MAP after each iteration using two types of the association matrix and their combination, with the best found values of  $\bar{p}$ , and two matrices with *all pairs* of probabilities, are plotted in Fig. 1. Combining visual and text modalities results in better performance than using either separately. In the runs where text from the topics description is used as the query (Fig. 1b), the difference in average precision is smaller, which is an expected result: the shots that are relevant because of the initial query text are put on top of the ranked list, and further search depends on this prior distribution by the nature of the Bayesian approach.

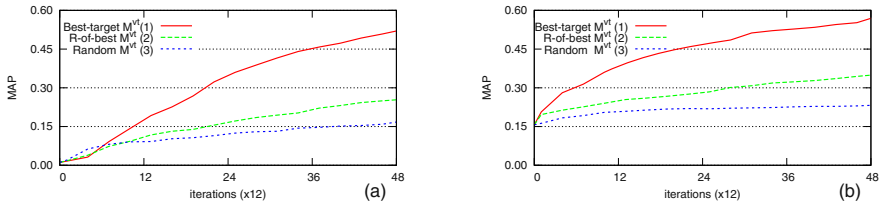


**Fig. 1.** MAP for different matrices vs. all pairs, without (a) and with (b) text-based prior distribution. In (b) the difference between curves (1) and (2) is not significant.

In the automated experiments the threshold  $1 - \alpha$  is such that on average 3% of possible values need to be stored. Keeping only significant  $P(\delta_i|T)$  in fact improves the search quality compared to the complete set of conditional probabilities both with visual and text-based matrices. This suggests that the probabilities replaced with the constant  $\bar{p}$  are indeed far from true similarities.

**The display update strategies.** The *Best-target display update* with an ad-hoc tuned value of  $\bar{p}$  offers great improvement over iterations, both when using the text priors and not. By making sure that the user does not see the same object twice, the danger of getting stuck in a local maximum is eliminated.

The *two non-deterministic strategies* perform not so well, especially when prior text information is used. The Non-deterministic methods perform on average 10 to 15 percent better if negative examples are *ignored* during the update of  $P(T)$ . As expected, the combination of Randomised and Best-target strategies (Random-of-Best) did better than the ‘pure’ Randomised. Still, uniform sampling of the more relevant subset of it, as done in Random-of-best, cannot beat the deterministic Best-target method. Sampling according to the estimated distribution of  $P(T)$  might be a better option. In Fig. 2 the best-performing combination is plotted for each display update strategy.



**Fig. 2.** MAP for different display update strategies without (a) and with (b) the prior text information.

**The prior distribution** based on text from the query description and words from speech transcripts provides overall better performance. Nevertheless, having little or no a priori information does not necessarily mean poor performance: Curve 1 in Fig. 1a for the method with no prior information available, reaches numbers comparable with the corresponding curve in Fig. 1b.

### 4.3 Live Experiments

In the *live* experiments, the search tasks have been performed by real users<sup>3</sup>. The data set contained about 32 000 key frames taken from 60 hours of news videos. We found high agreement between real users feedback and TREC relevance judgements (average among runs 75%), so our automated experiments can be viewed as a good approximation to real life (see [17] for an analysis of agreement between TREC assessors).

The set-up is similar to the automated experiments using the Best-target display update schema and text-based prior distribution  $P(T)$ . The user was allowed to see key frames (images), and not the corresponding videos. Only positive feedback from the user was taken into account. The resulting MAP at the

<sup>3</sup> 2 groups of 3 users to test 3 systems. All users are students of University of Twente aged between 19 and 26. Each search task took at most 15 minutes.

end of the live experiment evaluated by TREC is 0.245. For this run, 78% of the shots selected by the user were relevant according to TREC. At the same time, 48% of the relevant shots that have been displayed, were missed by our users. In the experiment that showed the user random screens (MAP 0.026), the number of missed shots was much lower (31%), as well as agreement with TREC (55%). The relevant documents are missed partially due to the fact that the user saw *still frames*, and not the *videos* themselves, but the difference in numbers between the runs suggests that relativity of the users' judgements (the user selects best of what is available and two users do not always agree) plays a role, too.

#### 4.4 Scalability of the Approach

The term 'scalability' denotes not only the possibility to run a retrieval system on a larger collection. The ability of a retrieval system to produce answers to the user's queries in a reasonable amount of iterations is at least as important.

We ran a number of automated experiments on a system consisting of 32 000 key frames from the TRECVID 03 data. After 48 iterations on the large collection, MAP of the best automated run is 0.44, compared to 0.58 achieved on the small collection. Note that half of the small collection were key frames relevant to one of the topics, whereas in the large collection only 6.5% of the key frames was relevant to one of the 25 topics. The execution time on the large collection, which is eleven times bigger than the small one, increased by factor 5 to 6.

### 5 Conclusions and Future Work

We found that feature normalisation and 'refinement' by way of replacing non-significant similarities with a constant which we propose, results in better search quality in both investigated feature spaces, text-based and visual-based. Using the association matrix as an index structure enables efficient combination of different modalities, such as visual information from key frames and transcripts of the speech occurring in video shots. Combining text and video (in the form of key frames) has positive effect on retrieval.

Organising the objects in a multimedia collection using the association matrix allows scalable implementation which is hard to achieve otherwise: computing similarities 'on the fly' is expensive in the sense of access time and/or computation effort, whereas keeping all pre-computed similarities is impractical from the storage point of view. Keeping only the significant similarities allows building an interactive content-based retrieval system that provides fast response time and good search quality on rather large image or video collections.

Text, in the form of speech transcripts of videos or annotations, is an important source of information about the multimedia content. When available, the text data should be used in combination with pictorial features, to improve the search results.

In the future we want to have the probabilities stored in the association matrix, to be updated by utilising the relevance judgements obtained from the

user's feedback. We are also going to investigate how to dynamically change the search strategy depending on user-system performance.

## References

1. *TRECVID 2003 Workshop, Notebook Papers*, 2003.
2. S. Aksoy and R. M. Haralick. Feature normalization and likelihood-based similarity measures for image retrieval. *Pattern Recognition Letters*, 22(5):563–582, 2001.
3. I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos. The bayesian image retrieval system, PicHunter: Theory, implementation, and psychophysical experiments. *IEEE Tran. On Image Processing*, 9(1):20–37, 2000.
4. H. Drucker, B. Shahrar, and D. C. Gibbon. Relevance feedback using support vector machines. In *Proc. 18<sup>th</sup> Int. Conf. on Machine Learning*, pages 122–129. Morgan Kaufmann, San Francisco, CA, 2001.
5. E.M.Voorhees, editor. *Proc. 10<sup>th</sup> Text Retrieval Conference, TREC-10*, 2002.
6. C. Faloutsos. *Searching Multimedia Databases By Content*. Kluwer Academic Publishers, Boston, USA, 1996.
7. M. Flickner, H. Sawhney, W. Niblack, and J. Ashley. Query by image and video content: the QBIC system. In *IEEE Computer*, volume 28, pages 310–315, 1995.
8. J. L. Gauvain, L. Lamel, and G. Adda. The LIMSI broadcast news transcription system. *Speech Communication*, 37(1-2):89–108, 2002.
9. D. Hiemstra. *Using language models for information retrieval*. PhD thesis, Centre for Telematics and Information Technology, University of Twente, 2001.
10. Y. Ishikawa, R. Subramanya, and C. Faloutsos. MindReader: Querying databases through multiple examples. In *Proc. 24<sup>th</sup> Int. Conf. Very Large Data Bases*, pages 218–227, 1998.
11. J. Laaksonen, M. Koskela, and E. Oja. PicSOM: Self-organizing maps for content-based image retrieval. In *Proc. of IJCNN'99*, Washington, D.C., USA, 1999.
12. S. E. Robertson. The probability ranking principle in IR. *Journal of Documentation*, 33(4):294–304, 1977.
13. J. J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The Smart Retrieval System: Experiments in Automatic Document Processing*, pages 313–323. Prentice Hall, 1971.
14. Y. Rui and T. Huang. Optimizing learning in image retrieval. In *Proc. IEEE int. Conf. On Computer Vision and Pattern Recognition*, June 2000.
15. S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *Proc. 9<sup>th</sup> ACM Int. Conf. on Multimedia*, pages 107–118. ACM Press, 2001.
16. N. M. Vasconcelos. *Bayesian Models for Visual Information Retrieval*. PhD thesis, Massachusetts Institute of Technology, 2000.
17. E. M. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. *Information Processing Management*, 36(5):697–716, 2000.
18. E. W. Weisstein. *CRC Concise Encyclopedia of Mathematics*. CRC Press, 2002.
19. T. Westerveld, A.P. de Vries, A.R. van Ballegooij, F.M.G. de Jong, and D. Hiemstra. A probabilistic multimedia retrieval model and its evaluation. *EURASIP Journal on Applied Signal Processing*, (2):186–198, February 2003.



# Salient Regions for Query by Image Content

Jonathon S. Hare and Paul H. Lewis

Intelligence, Agents, Multimedia Group,  
School of Electronics and Computer Science,  
University of Southampton,  
Southampton, SO17 1BJ,  
United Kingdom  
{jsh02r, phl}@ecs.soton.ac.uk

**Abstract.** Much previous work on image retrieval has used global features such as colour and texture to describe the content of the image. However, these global features are insufficient to accurately describe the image content when different parts of the image have different characteristics. This paper discusses how this problem can be circumvented by using salient interest points and compares and contrasts an extension to previous work in which the concept of scale is incorporated into the selection of salient regions to select the areas of the image that are most interesting and generate local descriptors to describe the image characteristics in that region. The paper describes and contrasts two such salient region descriptors and compares them through their repeatability rate under a range of common image transforms. Finally, the paper goes on to investigate the performance of one of the salient region detectors in an image retrieval situation.

## 1 Introduction

Much previous work in the field of content based retrieval has been based around the concepts of using global descriptors to describe the content of the image. More recently researchers have begun to realise that global descriptors are not necessarily good when it comes to describing the actual objects within the images and their associated semantics. Two approaches have grown from this realisation; firstly approaches have been developed whereby the image is segmented into multiple regions, and separate descriptors are built for each region; and secondly, the use of salient points has been suggested.

The first approach has been demonstrated to work [1], although it has a large problem - that of how to perform the segmentation. Over the years many techniques for performing image segmentation have been suggested, although none really solve the problem of linking the segmented region to the actual object that is being described. Indeed, this shows that the non-naive segmentation problem is not just a bottom-up image processing problem, but also a top-down problem that requires knowledge of the true object, before it can be successfully segmented.

The second approach avoids the problem of segmentation altogether by choosing to describe the image and its contents in an altogether different way. The use of saliency in computer vision has become quite widespread in recent years. Saliency is often used to provide the basis for a visual attention mechanism that reduces the need for computational resources [2]. Historically, saliency was described by the term ‘interest point detectors’, but use of the term ‘saliency’ has come about from the large amount of psychology-based work on selective visual attention. By using salient points within an image, it is possible to derive a compact image description based around the local attributes of the salient points. A number of different methods for finding salient points have been suggested, from the simple Harris’ & Stephens [3] corner detector, to wavelet based approaches [4,5,6], to methods based around image entropy [7,8]. Many previous approaches to using salient points have generated feature-vectors from pixel data in fixed-sized regions around the salient point, usually a 3x3 or 9x9 pixel neighbourhood centred on the point [5], although some of the modern state-of-the-art detectors find affine invariant regions and generate descriptors from within the region [9,10,11]. This paper compares and contrasts an extension to previous work in which the concept of scale is used in the selection of salient points (or rather salient regions), and the pixel content of the entire region content to build the feature vector of the local descriptor.

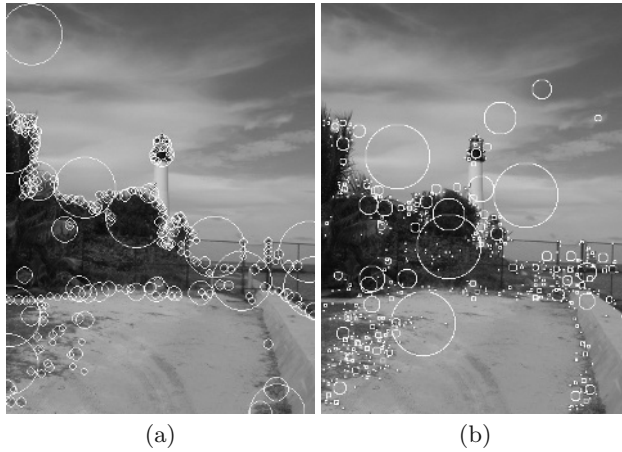
## 2 Salient Regions

### 2.1 Scale Saliency

The Scale-Saliency algorithm developed by Kadir and Brady [8,7] was based on earlier work by Gilles [12]. Gilles investigated salient local image patches or ‘icons’ to match and register two images (specifically aerial reconnaissance images). Gilles suggested that by extracting locally salient features from the pair of images and matching these, it would be possible to estimate the global transform between the two images. Gilles defined saliency in terms of local signal complexity or unpredictability. More specifically, he suggested the use of Shannon Entropy of local attributes to estimate the saliency. Basically, image segments with flatter intensity histogram distributions<sup>1</sup> tend to have higher signal complexity and thus higher entropy. Gilles method only worked at a single scale, and picked single salient points, rather than salient regions.

Kadir and Brady modified Gilles original algorithm to make it perform well on images other than those from aerial reconnaissance imagery. Essentially they changed the algorithm so that it detected salient regions at multiple scales by looking for self-similarity across scales. The modified algorithm located circular patches of the original image that were considered salient. The size of the patch was determined automatically by the multi-scale additions to Gilles algorithm.

<sup>1</sup> Kadir and Brady [8] note that the method is not limited to the intensity histogram and that it is equally possible to use a histogram from a different descriptor, such as colour or edge strength.



**Fig. 1.** (a) Salient regions found by the Scale-Saliency algorithm; (b) Salient regions found by from peaks in a difference-of-Gaussian pyramid

In addition Kadir and Brady developed a simple clustering algorithm to group together features within the  $\mathbb{R}^3$  space that have similar  $x$  and  $y$  location, and scale. Figure 1(a) illustrates the results of applying the algorithm to an image.

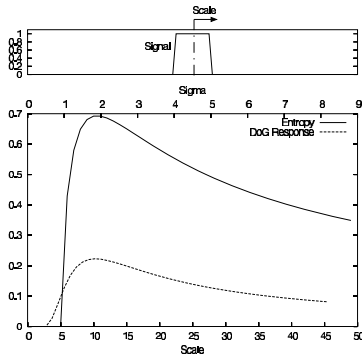
## 2.2 Peaks in the Difference-of-Gaussian Pyramid

We take the idea of using peaks in a difference-of-Gaussian pyramid from the work of Lowe [13,14] on object recognition using keypoints. Lowe has shown that by searching a difference-of-Gaussian pyramid for local peaks, both spatially and across scale, it is possible to select points robust to a range of projective transformations. The difference-of-Gaussian closely approximates the scale-normalised Laplacian-of-Gaussian [15,13],  $\sigma^2 \nabla^2 G$ . Mikolajczyk [16] showed that the minima and maxima of  $\sigma^2 \nabla^2 G$  produced the most stable interest points when compared to a range of other operators. Figure 1(b) illustrates the results of finding peaks in a difference-of-Gaussian pyramid.

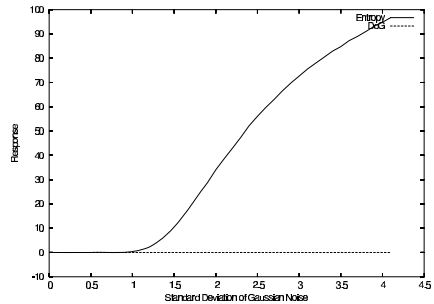
## 2.3 Comparison of Salient Region Methods

Both of the methods for selecting salient regions described above are quite similar. For example, when the response of a difference-of-Gaussian filter is large, we would also expect the entropy taken over the same area as the filter to be large. Note that the converse is not always true though - high entropy does not necessarily mean that there would be a large difference of gaussian response. This is illustrated in Figure 2.

One problem with entropy is that it is very sensitive to noise. This is especially so at small scales, where there are relatively few pixels to sample and estimate the probability density function from, in order to estimate the entropy.



**Fig. 2.** Entropy and difference-of-Gaussian (ratio of  $\sigma$ 's = 1 : 1.6, smaller  $\sigma$  is shown on the top x-axis) response versus scale to a one-dimensional signal as illustrated in the top diagram. The centre of the DoG and Entropy mask are kept at a constant position relative to the signal (shown by the dashed line). The graph illustrates how the response functions behave in a similar manner across scale-space



**Fig. 3.** Response of Entropy and difference-of-Gaussian functions to a constant signal with increasing amounts of zero-mean additive Gaussian noise. The DoG response stays stationary, whilst the Entropy response increases with noise

The difference-of-Gaussian is much less sensitive to noise due to the smoothing effect of the Gaussians. This is illustrated in Figure 3.

The remainder of this section is devoted to objectively comparing the stability of the two salient region detectors.

**Repeatability.** We take the measure of repeatability of interest points from Schmid *et al* [17]. The concept of repeatability is described below together with some results.

*Repeatability Criterion.* Repeatability is a measure of how independent an interest point detector is to the imaging conditions, i.e. camera parameters - position relative to the scene, zoom, etc. 3D points detected in one image should also be detected at approximately the same locations in subsequent images. Given a point  $X$  in 3D space and two projection matrices,  $P_1$  and  $P_2$ , the projections of  $X$  in two images  $I_1$  and  $I_2$  are given by  $p_1 = P_1X$  and  $p_2 = P_2X$  respectively. The point  $p_1$ , detected in image  $I_1$ , is repeated if the corresponding point  $p_2$  is detected in image  $I_2$ . In order to estimate the repeatability, a unique relation between the points  $p_1$  and  $p_2$  has to be found. In the case of a planar scene, points in one image are related to points in a second image by a planar homography:  $p_2 = Hp_1$ .

The percentage of points that are repeated with respect to the total number of detected points is called the repeatability rate. In general, a point is not repeated at exactly the same position as given by  $Hp_1$ , but in a small neighbourhood of that point. Denoting the size of the neighbourhood by  $\epsilon$ , we can define the  $\epsilon$ -*repeatability*. Interest points that cannot be observed in both images will corrupt the repeatability measure, thus only points in the common part of the scene are used to calculate the repeatability. The common part of the scene is defined by the homography, thus points  $\tilde{p}_1$  and  $\tilde{p}_2$  which lie in the common parts of images  $I_1$  and  $I_2$  are defined by  $\{\tilde{p}_1\} = \{p_1 | Hp_1 \in I_2\}$  and  $\{\tilde{p}_2\} = \{p_2 | H^{-1}p_2 \in I_1\}$ . The set of point pairs  $(\tilde{p}_1, \tilde{p}_2)$  that correspond within an  $\epsilon$ -neighbourhood is  $D(\epsilon) = \{(\tilde{p}_2, \tilde{p}_1) | dist(\tilde{p}_2, H\tilde{p}_1) < \epsilon\}$ .

As the number of detected points in the two images may be different, the repeatability rate is defined as:

$$r(\epsilon) = \frac{|D(\epsilon)|}{\min(|\{\tilde{p}_1\}|, |\{\tilde{p}_2\}|)}. \quad (1)$$

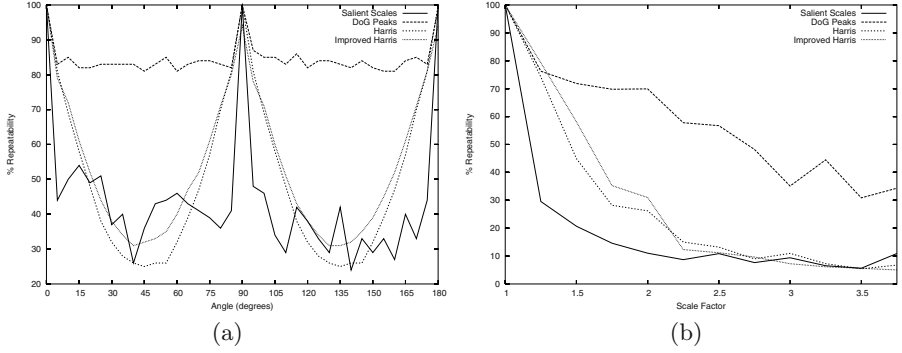
*Repeatability Results.* Using the repeatability criterion, we investigated the robustness of the two salient region descriptors to image rotation and scaling. The rotation and scaling were performed digitally, using bilinear interpolation. As a baseline, we also calculated the repeatability of the well-known Harris corner detector (using a  $[-2 \ -1 \ 0 \ 1 \ 2]$  kernel), and an improved version of the Harris detector that calculates the derivatives more precisely by replacing the  $[-2 \ -1 \ 0 \ 1 \ 2]$  kernel with one calculated from the derivatives of a Gaussian ( $\sigma = 1.0$ ).

Figure 4(a) illustrates the results of repeatability against rotation angle, averaged over all of the images in the dataset, and Figure 4(b) illustrates the variation in repeatability over a range of image scales, again averaged over all the images in the dataset. The results show that the salient regions detected by finding peaks in the difference-of-Gaussian pyramid are by far the most stable to both rotation and scaling. The salient-scales algorithm performs more-or-less on a par with the Harris detector. Unfortunately, whilst the salient-scales algorithm should be robust to both scaling and rotation, in practice it is affected by discretisation of the digital raster, especially at small scales. Also, we have found that the clustering part of the salient scales algorithm does little to help its stability.

### 3 Query by Image Content Using Salient Regions

In previous work by Sebe *et al* [5], the use of salient point detectors for content-based retrieval was shown to have better performance than when using global descriptors. In this section we describe a new metric for measuring the performance of content-based retrieval based on salient points, and illustrate it with some preliminary results that show that the performance when using salient regions is indeed better than when using global descriptors.

In order to facilitate the testing of the use of salient regions for content-based retrieval, we have developed a system that returns the  $N$  closest matches



**Fig. 4.** Repeatability rate for image rotation (a), and for scale change (b).  $\epsilon = 1.5$  in both cases

to a given query image. The system enables queries to be made using either global descriptors or a descriptor based on salient regions. Following Sebe *et al*, we fix the number of salient regions to 50 per image. In the case of global descriptors, the distance between two images,  $I_1$  and  $I_2$ , is given by the euclidean distance between the feature descriptors,  $\mathbf{F}_1$  and  $\mathbf{F}_2$ :

$$D_E(\mathbf{F}_1, \mathbf{F}_2) = \|\mathbf{F}_1 - \mathbf{F}_2\| = \sqrt{\sum_{i=1}^K |\mathbf{F}_{1i} - \mathbf{F}_{2i}|^2}, \quad (2)$$

where  $K$  is the number of elements in the feature descriptors. In the case of matching using salient regions, the distance between two images is given by a linear summation of the closest matching feature vector in the second image for each feature vector in the first image. Denoting the set of  $M$  feature vectors in images  $I_1$  and  $I_2$  as  $\{\mathbf{F}_1\}$  and  $\{\mathbf{F}_2\}$ :

$$D_{\text{salient}}(\{\mathbf{F}_1\}, \{\mathbf{F}_2\}) = \sum_j^M \min_k (D_E(\{\mathbf{F}_1\}_j, \{\mathbf{F}_2\}_k)), \quad (3)$$

where  $\{\mathbf{F}_1\}_j$  refers to the  $j$ th feature vector of image  $I_1$  and  $\{\mathbf{F}_2\}_k$  refers to the  $k$ th feature vector of image  $I_2$ .

### 3.1 Semantic Relevance

The problem with global descriptors is that they cannot fully describe all parts of an image having different characteristics. The use of salient regions aims to avoid this problem by developing descriptors that do capture the characteristics of each part of the image. Given this aim, it should not be unreasonable to expect that an image description generated from salient regions will be *better* than an image described wholly by a global descriptor. In order to test this we

**Table 1.** Averaged Semantic Relevance for queries based on the Rank 1 result image and the closest 5 result images

	Rank 1 Result Image		Averaged Top 5 Result Images	
Feature Type	DoG Peaks	Global	DoG Peaks	Global
RGB Histogram	42.1%	37.6%	51.0%	45.6%
HSI Histogram	45.2%	36.9%	50.4%	49.6%
Mono Histogram	31.6%	36.9%	42.3%	45.0%
HU Moment	41.1%	22.6%	52.4 %	39.5%
RGB Colour Moment	33.7%	24.1%	41.9%	35.4%
HSI Color Moment	34.9%	30.2%	43.5%	40.5%

have developed a metric that uses semantically marked images as ground-truth against the results from our retrieval system.

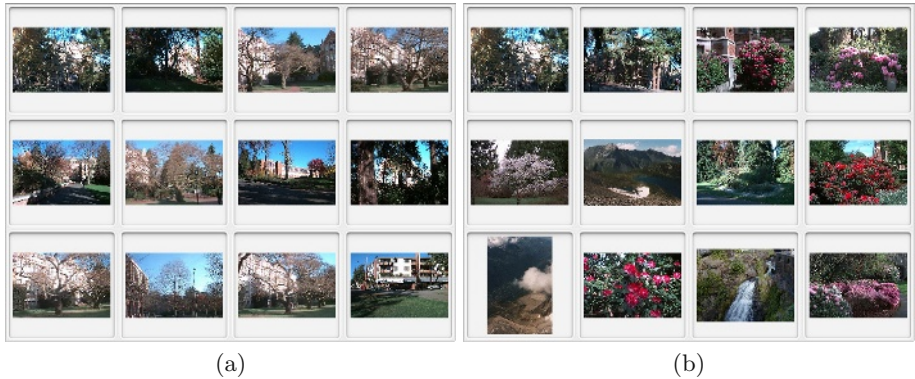
The University of Washington Ground Truth Dataset [18] contains a large number of images that have been semantically marked up. For example an image may have a number of labels describing the image content, such as “trees”, “bushes”, “clear sky”, etc. Given a query image with a set of labels, we should expect that the images returned by the retrieval system should have the same labels as the query image. Let  $A$  be the set of all labels from the query image, and  $B$  be the set of labels from a returned image. We then define the semantic relevance,  $R$ , of the query to be:

$$R = \frac{|A \cap B|}{|A|} \quad (4)$$

This implies that if all the labels in set  $A$  exist in set  $B$  then the semantic relevance will be 100%, and if only half of the labels in set  $A$  exist in set  $B$  then the semantic relevance will be 50%.

### 3.2 Results

We used all of the semantically marked images from the Washington dataset to form our test set. Taking each image in the test set in turn as a query, we calculated the distance to each of the other images in the test set using a range of feature types. We then calculated the semantic relevance for the rank one image (the closest image, not counting the query image), and we also calculated the averaged semantic relevance over the closest 5 images. The results of this are shown in Table 1. The table shows that the use of salient regions does indeed produce better semantic relevance than using global descriptors, although we believe that there is still scope for improvement of the semantic relevance from the salient regions. We believe that using a single feature type to describe a salient region (or indeed the whole image) is not sufficient. For example, the RGB histogram that represents a “blue sky” semantic label may be very similar to the histogram representing the “water” label. In our future work we hope to show it is possible to improve the semantic relevance of queries using salient regions by fusing multiple feature descriptors. Figure 5 illustrates the differences



**Fig. 5.** Example Retrieval: (a) shows the results of a query using the Difference of Gaussian salient region method, and (b) shows the results of the same query with the Global method. In both cases, RGB Histograms are used as the feature descriptor and the first image shown is the query image

between a query based on a global RGB-Histogram descriptor, versus multiple RGB-Histogram descriptors based around salient regions found from the peaks in the difference-of-Gaussian pyramid.

## 4 Conclusions and Future Work

In this paper, we have illustrated the concept of using peaks in a difference-of-Gaussian pyramid to select scale-invariant salient regions. We have shown that peaks in the difference-of-Gaussian pyramid are robust to a range of transformations, and that they perform better than an alternative approach to finding salient regions based on image entropy.

We have also demonstrated the concept of using salient regions for content-based retrieval. We have introduced a new metric, which we have termed *semantic relevance*, for the measurement of the relevance of a semantically marked result image from a semantically marked query image.

Our results have shown that the use of salient regions for content-based retrieval produces better semantic relevance than global descriptors. However, we note that it should be possible to improve these results even more by the use of better feature descriptors.

As previously mentioned, our future plans are to use the fusion of multiple features to try and improve the semantic relevance. We also plan to extend our system to use a better distance metric, such as the Mahalanobis distance,  $D_M$ .

**Acknowledgements.** We are grateful to the EPSRC and Motorola UK Research Laboratory for their support of this work.



## References

- [1] Carson, C., Thomas, M., Belongie, S., Hellerstein, J.M., Malik, J.: Blobworld: Image segmentation using expectation-maximization and its application to image querying. In: Third International Conference on Visual Information Systems, Springer (1999)
- [2] Itti, L., Koch, C.: Computational modelling of visual attention. *Nat. Rev. Neurosci.* **2** (2001) 194–203
- [3] Harris, C., Stephens, M.: A combined corner and edge detector. In Mathews, M.M., ed.: Proceedings of the 4th ALVEY vision conference, University of Manchester, England (1988) 147–151
- [4] Shokoufandeh, A., Marsic, I., Dickinson, S.: View-based object recognition using saliency maps. *Image Vis. Comput.* **17** (1999) 445–460
- [5] Sebe, N., Tian, Q., Loupias, E., Lew, M., Huang, T.: Evaluation of salient point techniques. *Image and Vision Computing* **21** (2003) 1087–1095
- [6] Sebe, N., Lew, M.S.: Comparing salient point detectors. *Pattern Recognition Letters* **24** (2003) 89–96
- [7] Kadir, T.: Scale, Saliency and Scene Description. PhD thesis, University of Oxford, Department of Engineering Science, Robotics Research Group, University of Oxford, Oxford, UK (2001)
- [8] Kadir, T., Brady, M.: Saliency, scale and image description. *Int. J. Comput. Vis.* **45** (2001) 83–105
- [9] Tuytelaars, T., Gool, L.V.: Content-based image retrieval based on local affinity invariant regions. In: Third International Conference on Visual Information Systems. (1999) 493–500
- [10] Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: International Conference on Computer Vision. (2003)
- [11] Obdrzalek, S., Matas, J.: Image retrieval using local compact dct-based representation. In: DAGM-Symposium 2003. (2003) 490–497
- [12] Gilles, S.: Robust Description and Matching of Images. PhD thesis, University of Oxford (1998)
- [13] Lowe, D.: Distinctive image features from scale-invariant keypoints. To appear in *International Journal of Computer Vision* (2004)
- [14] Lowe, D.: Object recognition from local scale-invariant features. In: Proc. of the International Conference on Computer Vision ICCV, Corfu (1999) 1150–1157
- [15] Marr, D.: VISION: A computational Investigation into Human Representation and Processing of Visual Information. W. H. Freeman and Company (1982)
- [16] Mikolajczyk, K.: Detection of local features invariant to affine transformations. PhD thesis, Institut National Polytechnique de Grenoble, France (2002)
- [17] Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest detectors. *Int. J. Comput. Vis.* **37** (2000) 151–172
- [18] University of Washington: Ground truth image database.  
<http://www.cs.washington.edu/research/imagedatabase/groundtruth/>  
 (2004)

# Evaluation of Texture Features for Content-Based Image Retrieval

Peter Howarth and Stefan Rüger

Department of Computing, Imperial College London,  
South Kensington Campus, London SW7 2AZ  
{[peter.howarth](mailto:peter.howarth@imperial.ac.uk), [s.rueger](mailto:s.rueger@imperial.ac.uk)}@imperial.ac.uk

**Abstract.** We have carried out a detailed evaluation of the use of texture features in a query-by-example approach to image retrieval. We used 3 radically different texture feature types motivated by i) statistical, ii) psychological and iii) signal processing points of view. The features were evaluated and tested on retrieval tasks from the Corel and TRECVID2003 image collections. For the latter we also looked at the effects of *combining* texture features with a colour feature.

## 1 Introduction

Texture is a key component of human visual perception. Like colour, this makes it an essential feature to consider when querying image databases. Everyone can recognise texture but, it is more difficult to define. Unlike colour, texture occurs over a region rather than at a point. It is normally defined purely by grey levels and as such is orthogonal to colour. Texture has qualities such as periodicity and scale; it can be described in terms of direction, coarseness, contrast and so on [1]. It is this that makes texture a particularly interesting facet of images and results in a plethora of ways of extracting texture features. To enable us to explore a wide range of these methods we chose three very different approaches to computing texture features: The first takes a statistical approach in the form of co-occurrence matrices, next the psychological view of Tamura's features and finally signal processing with Gabor wavelets.

Our study is the first to focus an evaluation of texture features on the whole image, and to tailor features for optimum retrieval performance in this context. The majority of original papers devising or evaluating texture features used classification or segmentation tasks to measure performance [2,3,4,5]. Both of these tasks are significantly different to the problems faced in image retrieval where one looks at generic queries for an entire picture. Real pictures are made up of a patchwork of differing textures rather than the uniform texture images often used in studies, such as the ones taken from Brodatz's photo book [6]. To that effect we suggest encoding texture in terms of joint histograms of low dimensional texture characteristics over the image in the same way 3D colour histograms are computed, we have called this a Tamura image. Throughout our work we have considered how best to cope with varying image sizes, scales, formats and orientations.

In the next section we look at the features we have chosen and how they are computed. Sect. 3 then describes the image libraries and similarity measures we used for evaluation. Sect. 4 presents our initial results on a training set and suggests modifications and parameters that we found gave the best retrieval performance. A larger performance comparison is carried out on the TRECVID2003 data set. Finally, Sect. 5 concludes the paper and outlines further work.

## 2 Texture Features

### 2.1 Co-occurrence

Statistical features of grey levels were one of the earliest methods used to classify textures. Haralick [7] suggested the use of grey level co-occurrence matrices (GLCM) to extract second order statistics from an image. GLCMs have been used very successfully for texture classification in evaluations [2].

**Table 1.** Features calculated from the normalised co-occurrence matrix  $P(i, j)$

Feature	Formula
Energy	$\sum_i \sum_j P^2(i, j)$
Entropy	$\sum_i \sum_j P(i, j) \log P(i, j)$
Contrast	$\sum_i \sum_j (i - j)^2 P(i, j)$
Homogeneity	$\sum_i \sum_j \frac{P(i, j)}{1 +  i - j }$

Haralick defined the GLCM as a matrix of frequencies at which two pixels, separated by a certain vector, occur in the image. The distribution in the matrix will depend on the angular and distance relationship between pixels. Varying the vector used allows the capturing of different texture characteristics. Once the GLCM has been created, various features can be computed from it. These have been classified into four groups: visual texture characteristics, statistics, information theory and information measures of correlation [7,3]. We chose the four most commonly used features, listed in Table 1, for our evaluation.

### 2.2 Tamura

Tamura et al took the approach of devising texture features that correspond to human visual perception [1]. They defined six textural features (coarseness, contrast, directionality, line-likeness, regularity and roughness) and compared them with psychological measurements for human subjects. The first three attained very successful results and are used in our evaluation, both separately and as joint values.

**Coarseness** has a direct relationship to scale and repetition rates and was seen by Tamura et al as the most fundamental texture feature. An image will

contain textures at several scales; coarseness aims to identify the largest size at which a texture exists, even where a smaller micro texture exists. Computationally one first takes averages at every point over neighbourhoods the linear size of which are powers of 2. The average over the neighbourhood of size  $2^k \times 2^k$  at the point  $(x, y)$  is

$$A_k(x, y) = \frac{1}{2^{2k}} \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} f(i, j) .$$

Then at each point one takes differences between pairs of averages corresponding to non-overlapping neighbourhoods on opposite sides of the point in both horizontal and vertical orientations. In the horizontal case this is

$$E_{k,h}(x, y) = |A_k(x + 2^{k-1}, y) - A_k(x - 2^{k-1}, y)| .$$

At each point, one then picks the best size which gives the highest output value, where  $k$  maximizes  $E$  in either direction. The coarseness measure is then the average of  $S_{\text{opt}}(x, y) = 2^{k_{\text{opt}}}$  over the picture.

**Contrast** aims to capture the dynamic range of grey levels in an image, together with the polarisation of the distribution of black and white. The first is measured using the standard deviation of grey levels and the second the kurtosis  $\alpha_4$ . The contrast measure is therefore defined as

$$F_{\text{con}} = \sigma / (\alpha_4)^n \quad \text{where} \quad \alpha_4 = \mu_4 / \sigma^4 ,$$

$\mu_4$  is the fourth moment about the mean and  $\sigma^2$  is the variance. Experimentally, Tamura found  $n = 1/4$  to give the closest agreement to human measurements. This is the value we used in our experiments.

**Directionality** is a global property over a region. The feature described does not aim to differentiate between different orientations or patterns, but measures the total degree of directionality. Two simple masks are used to detect edges in the image. At each pixel the angle and magnitude are calculated. A histogram,  $H_d$ , of edge probabilities is then built up by counting all points with magnitude greater than a threshold and quantising by the edge angle. The histogram will reflect the degree of directionality. To extract a measure from  $H_d$  the sharpness of the peaks are computed from their second moments.

**Tamura Image** is a notion where we calculate a value for the three features at each pixel and treat these as a spatial joint coarseness-contrast-directionality (CND) distribution, in the same way as images can be viewed as spatial joint RGB distributions. We extract colour histogram style features from the Tamura CND image, both marginal and 3D histograms. The regional nature of texture meant that the values at each pixel were computed over a window. A similar 3D histogram feature is used by MARS [8].

### 2.3 Gabor

One of the most popular signal processing based approaches for texture feature extraction has been the use of Gabor filters. These enable filtering in the frequency and spatial domain. It has been proposed that Gabor filters can be used to model the responses of the human visual system. Turner [9] first implemented this by using a bank of Gabor filters to analyse texture. A bank of filters at different scales and orientations allows multichannel filtering of an image to extract frequency and orientation information. This can then be used to decompose the image into texture features.

Our implementation is based on that of Manjunath et al [10,11]. The feature is computed by filtering the image with a bank of orientation and scale sensitive filters and computing the mean and standard deviation of the output in the frequency domain.

Filtering an image  $I(x, y)$  with Gabor filters  $g_{mn}$  designed according to [10] results in its Gabor wavelet transform:

$$W_{mn}(x, y) = \int I(x, y) g_{mn}^*(x - x_1, y - y_1) dx_1 dy_1$$

The mean and standard deviation of the magnitude  $|W_{mn}|$  are used to form the feature vector. The outputs of filters at different scales will be over differing ranges. For this reason each element of the feature vector is normalised using the standard deviation of that element across the entire database.

## 3 Experimental Set Up

We followed a two-stage approach: Initial evaluation and modifications to the features were tested using a carefully selected subset of the Corel image library and the vector space similarity measure. We then ran larger tests on the TRECVID2003 data using the  $k$ -nearest neighbour measure ( $k$ -nn). We have a baseline for evaluation from previous work with the TREC dataset for which  $k$ -nn has consistently proved the best retrieval method.

**Image Collections.** We selected 6,192 images from the Corel collection to give 63 categories that were visually similar internally, but different from each other [12]. A set of 630 single-image category queries was executed to test performance across all categories. Relevance judgments on the retrieved images were based on the categorisation. The results shown in Section 4 are the mean average precision (m.a.p.).

A second larger image collection was used to give a more realistic performance comparison. This comprised of 32,318 key-frames from TRECVID2003 collection [13]. The search task specified for TRECVID2003 consisted of 25 topics, for each topic a few example images were given as a query. The published relevance judgments for these topics were used to evaluate the retrieval performance for different features and combinations of features.

**Similarity Measures.** Distances between feature vectors were calculated using the Manhattan metric. The resultant distances were then median normalised to give even weighting when combined. The plain vector space model was used for retrieval on the Corel data set as these involved only simple 1-image queries.

For querying the TREC data a version of the distance weighted  $k$ -nn approach was used [14], with  $k = 40$ . Positive examples ( $P$ ) are supplied as the query and negative examples ( $N$ ) randomly selected from the collection. To rank an image  $i$  in the collection we identify those images in  $P$  and  $N$  that are amongst the  $k$ -nearest neighbours of  $i$ . Using these neighbours we determine the dissimilarity:

$$D(i) = \frac{\sum_{n \in N} d^{-1}(i, n)}{\sum_{p \in P} d^{-1}(i, p)}$$

## 4 Evaluation and Results

For each feature we evaluated performance in the configuration described in Sect. 2. Ideas to improve performance were devised and evaluated. The general themes considered were how best to represent an entire image, how to accommodate differing sizes and scale of images and how to cope with the regional qualities of textures. These evaluations were run on the Corel data. Paired  $t$ -tests were carried out to check whether results were statistically significant at  $\alpha = 0.05$ .

The best performing features from the initial evaluation were then tested on the TRECVID2003 data set. Tests were run with each texture feature combined with a high performing colour feature.

### 4.1 Co-occurrence

The two main variables when creating a GLCM are the number of quantisation levels and the vector. We decided to use four vector angles: 0, 45, 90, 135 and four distances. This could be used to calculate up to sixteen GLCMs. However, as the statistics are not invariant under rotation we also tried summing the four angles at each distance into a single matrix. GLCMs can be made symmetrical by including the reverse vector; symmetric and asymmetric matrices were tested. The number of quantisation levels dictate the size of matrix and density of the matrix. This may become a problem with small images or tiles. The effect of varying quantisation between 4 and 64 levels was tried. Features were calculated for whole and tiled images.

Preliminary results showed that distances between 1 and 4 pixels gave the best performance. There was no significant difference between symmetrical and asymmetric matrices. Tiling of the image gave a large increase in retrieval which flattened out by  $9 \times 9$  tiles. The results in Table 2 are for  $7 \times 7$  tiles. Similarly increasing quantisation improves performance. The concatenated features (cat) gave better results at all points than the rotationally invariant summed matrices (sum). The best feature was homogeneity with a m.a.p. of 12.2%.

**Table 2.** Co-occurrence features — mean average precision retrieval

Feature	Quantisation				
	4	8	16	32	64
Energy: cat	7.63%	8.09%	9.30%	9.85%	9.54%
Energy: sum	7.04%	7.79%	8.85%	9.19%	8.96%
Entropy: cat	8.12%	9.22%	10.41%	11.09%	11.36%
Entropy: sum	7.54%	8.76%	9.79 %	10.37%	10.70%
Contrast: cat	8.46%	8.51%	8.35%	8.29%	8.28%
Contrast: sum	7.83%	7.85%	7.65%	7.59%	7.57%
Homogeneity: cat	9.17%	10.18%	11.16%	11.83%	12.19%
Homogeneity: sum	8.50%	9.52%	10.39%	10.93%	11.26%

4.2 Tamura

When calculating standard Tamura features for whole or tiled images the main variable is the  $k$  value for coarseness. This effect of varying this, and the number of tiles, can be seen in Table 3. The dashes in the table are where the image size resulting from tiling meant that the  $k$  value was too large to be used because of the border needed.

With the histogram features the main variable to evaluate was the window size. Coarseness can be calculated at a pixel level. However, both the directionality and contrast features operate over a region. A large window would smear the feature and lose resolution; conversely a small window may invalidate the statistical features, particularly if the directionality histogram is too sparsely populated. To evaluate this the features were run over several window sizes, creating a histogram for each feature.

A little surprisingly initial results showed that increasing the  $k$  value for coarseness reduced the performance — the optimum value was 2. This may be due to the large borders necessary for higher values of  $k$ . However, it is more likely caused by the nature of textures in images and the way the algorithm averages the  $2^k$  values. There are unlikely to be textures with a coarseness of 64 or 32 pixels in a normal image. The algorithm may still detect noise at this dimension, biasing the average value of the feature. A change to the algorithm was made so that it took the values of  $k$  rather than  $2^k$  — effectively introducing a logarithmic scaling of the coarseness and giving less influence to the larger scales. This gave a significant increase in performance for the histogram, from 6.1% to 10.1%, but no improvement when applied to the standard feature.

Performance of the directionality feature was poor. A detailed look at the operation of the algorithm showed that this was largely due to the sparse population of the histogram and subsequent difficulty in calculating valid variance of its peaks. Several options for improvement were tried including calculating global variance of the histogram and using entropy. The latter gave a substantial improvement, from 6.6% to 9.7%, for the standard feature but negligible effect on the histogram.

**Table 3.** Tamura features — mean average precision retrieval

Feature	Standard features					Histogram features			
	Tiling					Window size			
	1x1	3x3	5x5	7x7	9x9	2	4	8	16
Contrast	3.24%	6.08%	7.20%	8.07%	8.03%	5.96%	6.71%	7.01%	6.92%
Directionality: peak finding	2.91%	4.16%	5.02%	5.79%	6.64%	5.39%	5.59%	5.57%	4.93%
Directionality: entropy	2.74%	5.35%	7.45%	8.93%	9.73%	4.89%	4.37%	5.24%	5.43%
Coarseness-2: $2^k$	4.42%	8.33%	9.48%	9.87%	9.91%	6.90%	5.99%	6.09%	6.01%
Coarseness-3: $2^k$	3.54%	7.57%	8.79%	9.19%	9.02%	6.52%	5.85%	5.96%	5.83%
Coarseness-4: $2^k$	3.49%	7.16%	7.68%	6.98%	—	6.12%	5.71%	5.64%	5.40%
Coarseness-5: $2^k$	3.25%	5.74%	—	—	—	—	—	—	—
Coarseness-6: $2^k$	2.92%	—	—	—	—	—	—	—	—
Coarseness-2: $k$	4.43%	7.96%	9.32%	9.57%	9.59%	6.44%	9.98%	9.83%	8.22%
Coarseness-3: $k$	3.91%	7.50%	8.92%	9.10%	8.94%	5.68%	10.08%	9.24%	7.93%
Coarseness-4: $k$	3.41%	6.95%	7.74%	7.15%	—	8.81%	9.33%	8.12%	7.67%

Finally the combined marginal and 3D histograms were evaluated using a window size of 8,  $k$  of 3 and entropy directionality. In addition a combined feature vector of the 3 standard features was evaluated. The m.a.p. results were: marginal histogram 12.0%, 3D histogram 13.7% and standard 14.3%. All gave a significant improvement over the single features.

### 4.3 Gabor

Sect. 2.3 describes the generation of this feature. However, there still remain questions over how to apply it to a heterogeneous set of images. The problems of scale, varying size and so on apply. The evaluation in [10] was applied to fixed tiles extracted from the Brodatz album. In [11] the feature was used successfully with aerial photographs split into a large number of fixed size tiles and then querying to find individual tiles. We decided to evaluate the feature in two configurations across a range of scale and orientation values. The first scaled the filter dictionary to the size of the image. This should scale the response so that the same image of different size gives a similar value. The second approach was to use a fixed size filter and apply this to a sliding window over the image.

Initial results showed that scaling the filter size gave much superior results to the sliding window approach. Tiling increased performance in a similar manner to the other features. The results shown in Table 4 are for  $7 \times 7$  tiling. The best performance is obtained from just 2 scales and 4 orientations. This was unexpected as most literature recommends 4 scales and 6 orientations. Looking at the filtered images indicated that, as for Tamura, this may be due to noise at coarser scales.

### 4.4 Evaluation Using TRECVID2003 Video Data

A range of the best performing features were run on the TRECVID2003 data and evaluated using the published relevance judgments. The queries were run singly



**Table 4.** Gabor wavelets — mean average precision retrieval

Scale	Orientation		
	3	4	6
2	13.1%	14.0%	13.9%
3	11.0%	11.4%	11.3%
4	10.8%	11.4%	11.2%

and then combined with a colour histogram feature, HSV [12]. The results are shown in Table 5. For comparison some features used for previous evaluations [12] gave m.a.ps of: HSV 1.9%, convolution 2.2% and variance 1.7%; random retrieval would give 0.26%.

In this evaluation the texture features performed extremely well in comparison with previous benchmarks. Gabor gave the best results, 3.9% or 15 times better than random retrieval. Of the Tamura features the best performing was the combined standard features. The top 3 performing texture features combined and giving a m.a.p of 4.22%.

Combining with the HSV feature improved average retrieval performance in all cases, but at an individual query level the benefits were both positive and negative. It is interesting that using simple combination of features gives varying degrees of improvement; being able to choose the optimum combination based on the query would be beneficial.

**Table 5.** TREC evaluation — mean average precision retrieval

Feature	Single	Combined with HSV
gabor-2-4	3.93%	4.31%
co-occurrence homogeneity	2.85%	3.03%
tamura standard all	2.57%	3.43%
tamura CND	1.65%	2.72%
tamura coarseness-2	0.97%	2.49%

## 5 Conclusions

We selected 3 different texture features, implemented and evaluated them. Both the evaluation and implementation focussed on query-by-example image retrieval rather than the usual classification task.

This led to some novel modifications to the Tamura features. We found that looking for large scale coarseness degraded performance, so we limited the range and used a logarithmic scale. An improvement in directionality performance over small window sizes was achieved by using an entropy measure rather than taking

the second moments of the peaks. We also encoded the features in terms of joint histograms, the overall performance of these was similar to the standard features.

To improve the retrieval with Gabor we scaled the filter size to that of the image, rather than using a fixed size filter. Rather unintuitively we found that fewer scales gave higher retrieval rates. Our tests of co-occurrence matrices showed a solid performance — as expected!

Our evaluation with TRECVID2003 data showed that the top 3 texture features performed better than previously used colour features. Combination with a colour feature boosted retrieval performance in all cases. Overall we have demonstrated that we have produced robust texture features for image retrieval.

We would like to carry out further evaluations on larger data sets, particularly investigating the interaction of different feature combinations. Finally, texture features have an advantage over colour features in that performance should be the same for monochrome images. It would be interesting to perform an evaluation on a library of black and white pictures.

**Acknowledgement.** This work was partially supported by the EPSRC, UK.

## References

1. Tamura, H., Mori, S., Yamawaki, T.: Textural features corresponding to visual perception. *IEEE Trans on Systems, Man and Cybernetics* **8** (1978) 460–472
2. Ohanian, P., Dubes, R.: Performance evaluation for four classes of textural features. *Pattern Recognition* **25** (1992) 819–833
3. Gotlieb, C.C., Kreyszig, H.E.: Texture descriptors based on co-occurrence matrices. *Computer Vision, Graphics and Image Processing* **51** (1990) 70–86
4. Jain, A.K., Farrokhnia, F.: Unsupervised texture segmentation using gabor filters. *Pattern Recognition* **23** (1991) 1167–1186
5. Randen, T., Husøy, J.H.: Filtering for texture classification: A comparative study. *IEEE Trans on Pattern Analysis and Machine Intelligence* **21** (1999) 291–310
6. Brodatz, P.: *Textures: A Photographic Album for Artists & Designers*. Dover (1966)
7. Haralick, R.: Statistical and structural approaches to texture. *Proceedings of the IEEE* **67** (1979) 786–804
8. Ortega, M., Rui, Y., Chakrabarti, K., Mehrotra, S., Huang, T.S.: Supporting similarity queries in MARS. In: *ACM Multimedia*. (1997) 403–413
9. Turner, M.: Texture discrimination by Gabor functions. *Biological Cybernetics* **55** (1986) 71–82
10. Manjunath, B., Ma, W.: Texture features for browsing and retrieval of image data. *IEEE Trans on Pattern Analysis and Machine Intelligence* **18** (1996) 837–842
11. Manjunath, B., Wu, P., Newsam, S., Shin, H.: A texture descriptor for browsing and similarity retrieval. *Journal of Signal Processing: Image Communication* **16** (2000) 33–43
12. Pickering, M., Rüger, S.: Evaluation of key-frame based retrieval techniques for video. *Computer Vision and Image Understanding* **92** (2003) 217–235
13. Alan Smeaton, W.K., Over, P.: TRECVID 2003 — An introduction. In: *TRECVID 2003 Workshop*. (2003) 1–10
14. Mitchell, T.M.: *Machine Learning*. McGraw Hill (1997)

# An Effective Approach Towards Content-Based Image Retrieval

Rokia Missaoui, M. Sarifuddin, and Jean Vaillancourt

Département d'informatique et d'ingénierie,  
Université du Québec en Outaouais  
C.P. 1250, Succ. B, Gatineau (Qc), Canada, J8X 3X7  
{rokia.missaoui, sarifuddin, jean.vaillancourt}@uqo.ca

**Abstract.** This paper describes a content-based approach to improve image retrieval effectiveness. First, we define two new measures for computing similarity among images based on color histograms, namely the dissimilitude distance  $DS^*$  and the similarity distance  $E$ . The latter is incorporated into the exponentiation part of the Gibbs distribution and into the generalized Dirichlet mixture, while the former is compared to five similarity measures:  $L_1$ ,  $L_2$  (Euclidean distance),  $E$  as well as Gibbs and Dirichlet distributions integrating the similarity measure  $E$ . Then, in order to overcome the limitations (and inappropriateness) of some previous information retrieval measures in evaluating the efficiency of an image retrieval process, three variants of a new effectiveness measure are proposed and experimented on an image collection for different similarity distances.

## 1 Introduction

Content-based image retrieval (CBIR) has emerged as an important area in computer vision, multimedia computing and databases. In order to make image databases easier to explore, we have developed a three-step process and a prototype for image mining and retrieval (see [5]). The main objectives of such a work are: (i) the development of an image feature extraction module, (ii) the design of a data mining tool dedicated to image clustering, classification and association rule generation, and (iii) the design of an image retrieval module which allows the identification of images that are similar to a given image query. In this paper, we limit ourselves to the third objective by describing the mechanisms put together to improve image retrieval effectiveness.

The rest of the paper is organized as follows. Section 2 presents a brief background on image color representation. Section 3 presents a new similarity distance  $E$  for image retrieval as well as its integration into two separate distributions, namely Gibbs distribution (more precisely, its exponentiation part) and generalized Dirichlet mixture [1]. A new retrieval measure is defined in Section 4 while details about the experimentation of our solution to improving image retrieval effectiveness is provided in Section 5.

## 2 Color Spaces

Most image retrieval systems follow the paradigm of representing images using a set of features, such as color, texture, shape and layout. Among these features, color is the most

frequently used visual property in content-based image retrieval because it is relatively robust, and invariant with respect to image size and orientation.

It is known that the  $RGB$  space is not perceptually uniform in the sense that color differences captured by the Euclidean distance, for example, in the three-dimensional  $RGB$  space do not correspond to color differences as perceived by humans. The CIE (*Commission Internationale de l'Eclairage*) has then defined two perceptually uniform or approximatively-uniform color spaces  $L^*a^*b^*$  and  $L^*u^*v^*$ . Further, the  $L^*C^*H^*$  (Lightness, Chroma, and Hue) and  $L^*t^*\theta^*$  ( $t$  = Chroma and  $\theta^*$  = Hue) color spaces have been defined as derivatives of  $L^*u^*v^*$  and  $L^*a^*b^*$  [4].

In our work we use 3-D  $L^*a^*b^*$  and  $L^*C^*H^*$  color spaces to represent and extract color properties of images. Any color in  $L^*a^*b^*$  space is represented in a cubic coordinate system of axes  $L^*$ ,  $a^*$ , and  $b^*$ . The mapping from  $L^*a^*b^*$  to  $L^*C^*H^*$  can be expressed in terms of polar coordinates with the perceived lightness and the psychometric correlates of chroma and hue angle using the following formula:

$$C_{ab}^* = \sqrt{a^{*2} + b^{*2}} \text{ and } H_{ab}^* = \tan^{-1} \left( \frac{b^*}{a^*} \right) \quad (1)$$

To get a good precision with a reasonably fair execution time, we apply Wand's quantization method [8] to 3-D color histogram  $L^*C^*H^*$ . We divide the hue angle  $H^*$  is divided into 17 colors ( $k = 0, 2, \dots, 16$ ), and each color is then split into 12 chroma ( $n = 0, 2, \dots, 11$ ) and 15 lightness values ( $m = 0, 2, \dots, 14$ ). For white and black images, color is split into 15 different lightness values.

Finally, each histogram is divided into  $3075^1$  color bins and represented by a vector  $\mathbf{V} = (V_{0,0,0}, V_{0,0,1}, \dots, V_{k,m,n}, \dots, V_{16,14,11})$  where  $V_{k,m,n}$  represents a color bin and stands for the percentage of pixels having the color, lightness and chroma in the quantization intervals  $k \triangleq h$ ,  $m \triangleq l$  and  $n \triangleq c$ .

### 3 Similarity Color Distance

Color-based similarity analysis can be conducted using either color vectors or color histograms and bins. Moreover, it can be conducted using either similitude, dissimilitude or both of them.

In case of similitude analysis, a simple metric distance  $L_q$  such as  $L_1$  (city-block,  $q = 1$ ) or  $L_2$  (Euclidean distance,  $q = 2$ ) can be used. However,  $L_1$  and  $L_2$  are not appropriate for the identification of dissimilitude.

$$L_q = \left( \sum_c |V^X(c) - V^Y(c)|^q \right)^{(1/q)} \quad (2)$$

As opposed to  $L_1$  and  $L_2$  distances which compute the difference between two histograms w.r.t. to color  $c$ , a metric called histogram intersection [7] is defined as the common proportion of color  $c$  in two histograms.

In this section we define a new similarity distance which takes into account both the dissimilitude and the similitude of two images  $X$  and  $Y$  with respect to a set of colors. We

<sup>1</sup> =  $(17 \times 15 \times 12) + 15$ .

first present a dissimilitude distance named  $DS^*$ , and then a similarity distance called  $E$ . Both distances are semi-metric ones. Finally, we integrate  $E$  into two distributions, namely Gibbs random field model and generalized Dirichlet mixture.

### 3.1 Dissimilitude Distance

We define the dissimilitude distance  $DS_c^*$  between two images  $X$  and  $Y$  with respect to color  $c$  as equal to  $D_c + L_c$  where  $D_c$  is an indicator of a potential absence of color  $c$  in one of the images, and  $L_c$  is the difference between the proportions of color  $c$  in images  $X$  and  $Y$ . The former helps discard some images that do not share the same set of colors with the query image.

$$L_c = |V^X(c) - V^Y(c)| \text{ and } D_c = \begin{cases} L_c & \text{if } L_c = \max(V^X(c), V^Y(c)) \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

where  $V^X(c)$  and  $V^Y(c)$  are the proportions of color  $c$  in histograms related to images  $X$  and  $Y$  respectively. It turns out that the dissimilarity distance  $D_c$  is equivalent to  $L_c$  (e.g.,  $L_1$ ) when color  $c$  is present in both histograms.

Based on the  $L^*C^*H^*$  color histogram, the color  $c$  corresponds to a color bin at coordinates  $k, m$ , and  $n$ . Using equation 3, the dissimilitude distance of a color bin and a color histogram can be expressed respectively by:

$$DS_{k,m,n}^* = \begin{cases} 2 \times L_{k,m,n} & \text{if } L_{k,m,n} = \max(V_{k,m,n}^X, V_{k,m,n}^Y) \\ L_{k,m,n} & \text{otherwise.} \end{cases} \quad (4)$$

$$DS^* = \left( \sum_{k=0}^{16} \sum_{m=0}^{14} \sum_{n=0}^{11} (DS_{k,m,n}^*)^q \right)^{1/q}. \quad (5)$$

Figures 1, 2 and 3 in Section 5 show the image retrieval output when  $L_1$ ,  $L_2$  and  $DS^*$  distances are used respectively. The leftmost top image on each figure represents the *image query* while the *retrieved images* are displayed by the system in a relevance ranking sequence (from top to bottom and left to right). One can see that  $DS^*$  leads to a smaller number of false alarms (i.e., number of irrelevant images included in the answer set) than  $L_1$  and  $L_2$ .

### 3.2 Similarity Distance

Similarity between two color histograms (related to images  $X$  and  $Y$ ) with respect to color  $c$  can be expressed by  $R_c$  as a ratio between the dissimilitude  $DS_c^*$  and the similitude  $S_c$  as follows:

$$R_c = \frac{DS_c^*}{S_c}, \text{ where } S_c = \min(V^X(c), V^Y(c)) \quad (6)$$

In order to better highlight similitude and dissimilitude between two images w.r.t. to color  $c$ , we propose a new similarity distance as a combination of the two components

$DS_{k,m,n}^*$  and  $S_{k,m,n}$ . The following two formulae express the similarity distance for color bins and color histograms respectively:

$$E_{k,m,n} = \begin{cases} DS_{k,m,n}^* (1 + \log(R_{k,m,n})) & \text{if } S_{k,m,n} > DS_{k,m,n}^* \\ DS_{k,m,n}^* & \text{otherwise} \end{cases} \quad (7)$$

where  $S_{k,m,n}$  is given by Equation 11

$$E = \left( \sum_{k=0}^{16} \sum_{m=0}^{14} \sum_{n=0}^{11} (E_{k,m,n})^q \right)^{1/q} \quad (8)$$

Figure 4 shows that  $E$  leads to a better retrieval output than  $L_1$ ,  $L_2$  and  $DS^*$ .

To further improve the effectiveness of the image retrieval procedure, we propose to incorporate the empirically superior distance measure  $E$  into two models that have been shown to possess powerful properties in image retrieval system in the past: the Gibbs random fields and Dirichlet mixture.

### 3.3 Similarity Distance and Gibbs Distribution

Gibbs distributions and Gibbs random fields are very popular in Statistical Physics and have been successfully used in image processing such as image enhancement, texture analysis, and image comparison [6].

A Gibbs random field (GRF) can be thought of as a random coloring of points on a lattice. It is therefore convenient to represent it mathematically as a family  $F$  of random variables taking values in a set  $\mathcal{S}$  and parameterized through each possible color configuration  $f$  on the lattice (which in our case is the two-dimensional support for the images). Usually, as is the case here, such a distribution is defined as follows :

$$P(f) = \frac{\exp - \frac{U(f)}{T}}{\sum_{f \in F} \exp - \frac{U(f)}{T}} \quad (9)$$

where the denominator is just a normalizing constant,  $T$  (a measure of the entropy of the distribution, usually referred to as the temperature) is set to value 1 for simplicity and  $U(f)$  is the so-called energy function, which in our case will take the form of a sum over neighboring configurations to  $f$  as defined by a prescribed neighborhood system  $\mathcal{N}$ .

$$U(f) = \sum_{C \in \mathcal{C}_\mathcal{N}} \left( \sum_{c \in \mathcal{C}_C(f)} V_C(c) \right) \quad (10)$$

where  $\mathcal{C}_\mathcal{N}$  is the set of clique types generated by the neighborhood system  $\mathcal{N}$ ,  $\mathcal{C}_C(f)$  is the set of instances of the clique type  $C$  in the lattice  $f$ , and  $V_C(\cdot)$  is the potential function associated with clique type  $C$  [3].

Using a similar reasoning as before, we can define the similitude measure between a color bin at position  $k, m, n$  in the histogram of images  $X$  and  $Y$  as follows:

$$S_{k,m,n} = \max \left( \min(V_{k,m,n}^X, V_{k,j,i}^Y), \min(V_{k,j,i}^X, V_{k,m,n}^Y) \right)_{j=(m-1,1,m+1)}^{i=(n-1,1,n+1)} \quad (11)$$

Since each histogram is quantized into  $k$  colors and each color is split according to  $n$  chroma values and  $m$  lightness values, the similitude of a color bin  $V^X(k, m, n)$  in the histogram of the query image can be computed with respect to the neighborhood ( $\mathcal{N} = 2$ ) of chroma and lightness of the color bin  $V^Y(k, m, n)$  related to a target image.

While dissimilitude is computed using Equation 4, the similarity distance  $E$  of color  $k$  and lightness  $m$  for any value of the chroma  $n$  (in the two histograms) is calculated using the following equation:

$$E_{k,m} = U(f) = \begin{cases} e^{-DS_{k,m}^* (1 + \log(\frac{DS_{k,m}^*}{S_{k,m}}))} & \text{if } S_{k,m} > DS_{k,m}^* > 0 \\ e^{-DS_{k,m}^*} & \text{otherwise} \end{cases} \quad (12)$$

$$U(X, Y) = \sum_{n=0}^{16} \sum_{m=0}^{14} E_{k,m} \quad (13)$$

where  $S_{k,m}$  and  $DS_{k,m}^*$  are given by:

$$S_{k,m} = \left( \min \left( \sum_{n=0}^{11} V_{k,m,n}^X, \sum_{n=0}^{11} V_{k,m,n}^Y \right)^q + \sum_{n=0}^{11} (S_{k,m,n})^q \right)^{(1/q)} \quad (14)$$

$$DS_{k,m}^* = \left( \left| \sum_{n=0}^{11} V_{k,m,n}^X - \sum_{n=0}^{11} V_{k,m,n}^Y \right|^q + \sum_{n=0}^{11} (DS_{k,m,n}^*)^q \right)^{(1/q)} \quad (15)$$

The motivation behind using an average value of chroma in Equation 12 is due to the fact that the variation of chroma does not lead to an abrupt change to a color perception while a lightness variation does.

Equation 12 is used to calculate the similarity between two histograms of images  $X$  and  $Y$  for a given color  $k$  and an identified lightness  $m$ , while Equation 13 is used to compute the distance between those histograms.

Using the distribution described by Equation 9 and taking Formula 13 as the energy function together with a default value of 1 for  $T$ , we define the probability that an image  $Y_j$  ( $j = 1, 2, \dots, J$ ) in the database be similar to a given query image  $X$  as:

$$P(X, Y_j) = \frac{\exp(-\sum_{k=0}^{16} \sum_{m=0}^{14} E_{k,m}^j)}{\sum_{j=1}^J \exp(-E^j)} \quad (16)$$

where  $E^j$  is the similarity distance between the query image  $X$  and image  $Y_j$  of the database.

Figure 5 illustrates image retrieval using similarity distance as an integration of  $E$  into the exponentiation part of Gibbs distribution.

### 3.4 Similarity Distance and Dirichlet Distribution

Let  $\mathbf{V} = (V_1, \dots, V_I)$  be a vector of positive random color variables and  $V_i$  the maximal probability value of the  $i^{th}$  color in the two histograms of  $X$  and  $Y$  (i.e.,  $V_i = \max(V_{k,m}^X, V_{k,m}^Y)$  or  $V_i = \max(V_k^X, V_k^Y)$ ) with  $\sum_{i=1}^I V_i < A$  and  $0 < V_i < 1$ .

Based on the generalized Dirichlet mixture [1], the joint density function is given by:

$$p(V_1, \dots, V_I) = \frac{\Gamma(|\alpha|)}{A^{|\alpha|} \prod_{i=1}^I \Gamma(\alpha_i)} \prod_{i=1}^I V_i^{\alpha_i - 1} \quad (17)$$

Vector  $\alpha = (\alpha_1, \dots, \alpha_I)$  can be perceived as a similarity distance for events governed by  $V_i$ , and hence  $\alpha_i$  can be instantiated to  $E_{k,m}$  (see Equation 12) for  $V_i = \max(V_{k,m}^X, V_{k,m}^Y)$ , or to  $E_k = \sum_m E_{k,m}$  for  $V_i = \max(V_k^X, V_k^Y)$ , where  $V_k^X = \sum_m \sum_n V_{k,m,n}^X$  and  $V_k^Y = \sum_m \sum_n V_{k,m,n}^Y$ . The parameter  $A$  is a constant.

Figure 6 shows that the integration of the similarity distance  $E$  into the Dirichlet distribution leads to the best retrieval effectiveness among the six alternatives considered.

## 4 Image Retrieval Effectiveness

The performance of an image retrieval system may be analyzed according to its accuracy and its efficiency. While the latter is estimated based on execution time and storage requirements, the former corresponds to system effectiveness in retrieving the images that are the most closely similar to the image query.

Indicators such as false alarm, false dismissal, precision and recall are commonly used for retrieval effectiveness computation. However, they do not really reflect the accuracy of the image retrieval system because the ranking of each displayed image is generally not taken into account. The normalized recall measure partially overcomes this limitation.

Faloutsos *et al.* [2] have defined a measure for evaluating the effectiveness of QBIC system. For each image query, the average rank (AVRR) of all relevant retrieved images is computed as well as the ideal average rank of relevant images (IAVRR). The formula assumes that the system returns all the  $P$  relevant images which, in the ideal case (IAVRR), occupy the first  $P$  positions. This effectiveness measure obviously takes into account the ranking of relevant images. However, it ignores the deviation between the ideal ranking and the actual ranking of a relevant image. For example, if the system returns images in a completely inverse order of the ideal ranking, the following formula returns a perfect effectiveness value ( $= 1$ ).

$$\text{Eff} = \frac{AVRR}{IAVRR}, \text{ where } IAVRR = \sum_{i=1}^P \frac{i}{P} \text{ and } AVRR = \sum_{i=1}^P \frac{r_i}{P} \quad (18)$$

where  $P$  is the total number of relevant images,  $i = (1, 2, \dots, P)$  is similarity image ranking by human expert judgement and  $r_i$  corresponds to system image ranking (in a decreasing relevance order).

In this section we propose a new effectiveness measure which overcomes the limitations indicated so far. Let  $P$  be the total number of relevant images in the image database,  $R$  the total number of retrieved images ( $R \geq P$ ) and  $P_R$  the accuracy ratio defined either by  $P_R = \frac{P}{R}$  or  $\frac{1}{1 + \log(\frac{R}{P})}$ , where  $0 \leq P_R \leq 1$ . We define the (actual) average rank as  $AVRR = \sum_{i=1}^P \frac{i}{P} + \sum_{i=1}^P \frac{|i - r_i|}{P}$  while the ideal average rank  $IAVRR$  is kept unchanged [2].



**Table 1.** Displayed images using L1, L2, DS\*, E, E+Exp and E+Dirichlet distances. False alarms are in **bold**.

Distance	Relevant image ranking (P=10)										R
Expert	1	2	3	4	5	6	7	8	9	10	10
$L_1$	1	2	5	4	3	6	<b>31</b>	<b>11</b>	<b>27</b>	<b>12</b>	31
$L_2$	1	2	<b>28</b>	2	10	<b>20</b>	<b>31</b>	5	<b>19</b>	<b>23</b>	31
$DS^*$	1	2	4	6	3	5	<b>26</b>	<b>13</b>	<b>23</b>	<b>12</b>	26
$E$	1	2	4	6	3	5	<b>24</b>	<b>14</b>	<b>20</b>	10	24
$E + Exp$	1	2	4	3	5	7	<b>12</b>	10	<b>17</b>	<b>11</b>	17
$E + Dirichlet$	1	2	4	3	6	5	7	8	<b>13</b>	10	13

**Table 2.** Retrieval effectiveness computation for L1, L2, DS\* ( $q = 1$ ), E, E+Exp and E+Dirichlet distances using seven measures.

Effectiveness method	Effectiveness retrieval of six different metric distances						
	Expert	$L_1$	$L_2$	$DS^*$	$E$	$E + Exp$	$E + Dirichlet$
Faloutsos	1.0	2.04	2.96	1.89	1.76	1.38	1.09
Kendall	1.0	0.689	0.533	0.667	0.667	0.778	0.867
Salton	1.0	0.995	0.991	0.996	0.997	0.998	0.999
Parkaew	1.0	0.883	0.678	0.865	0.863	0.908	0.927
$Eff_{ord}$	1.0	0.512	0.364	0.545	0.579	0.743	0.873
$Eff_{sys}(a)$	1.0	0.167	0.116	0.209	0.241	0.437	0.672
$Eff_{sys}(b)$	1.0	0.368	0.242	0.385	0.419	0.604	0.781

In the following we define two variants of our effectiveness measure. The first one, called  $Eff_{ord}$ , exploits ranking in a more accurate way than in [2] while the second, called  $Eff_{sys}$ , improves the former by taking into account the number  $R$  of retrieved images needed to display the  $P$  relevant ones. The last one can be split into two distinct variants depending on the value given to  $P_R$  (see above). The second variant is more appropriate when  $R \gg P$  while the first one performs better when  $R$  is a small multiple of  $P$ .

$$Eff_{ord} = \frac{\sum_{i=1}^P i}{\sum_{i=1}^P i + \sum_{i=1}^P |i - r_i|} \quad (19)$$

$$Eff_{sys} = \frac{P}{R} \frac{\sum_{i=1}^P i}{\sum_{i=1}^P i + \sum_{i=1}^P |i - r_i|} \quad (20a)$$

$$Eff_{sys} = \frac{1}{1 + \log(\frac{R}{P})} \frac{\sum_{i=1}^P i}{\sum_{i=1}^P i + \sum_{i=1}^P |i - r_i|} \quad (20b)$$

Our preliminary experiments show that for a given recall, the highest (respectively the lowest) precision occurs for  $E + Dirichlet$  (respectively  $L_2$ ).



**Fig. 1.** Image retrieval using L2 distance.



**Fig. 2.** Image retrieval using L1 distance.



**Fig. 3.** Image retrieval using  $DS^*$  distance ( $q=1$ ).



**Fig. 4.** Image retrieval using  $E$  distance ( $q=1$ ).



**Fig. 5.** Image retrieval using  $E+Exp$  distance.



**Fig. 6.** Image retrieval using  $E+Dirichlet$  distance.

## 5 Empirical Analysis

We have conducted the empirical analysis into two steps: (i) image retrieval using each one of the six distances on a collection of 1069 images, and (ii) effectiveness computation based on the expert's ranking of similar images and using seven effectiveness retrieval measures. For the first step, ten image queries were addressed to the database by four users (students and faculty members) and average execution time was computed. For each similarity measure, the system retrieves the  $R$  images needed to display the  $P$  (set to 10) relevant images. Some irrelevant images appear in the answer set (false alarms) while some relevant images will be missed (false dismissals).

While the two steps aim at analyzing the retrieval effectiveness of each one of the distances, the second step helps identify the behavior of the newly proposed effectiveness measures, namely  $Eff_{ord}$  and  $Eff_{sys}$ .

Figures 1 through 6 show the images ranked by the system (from top to bottom and from left to right) when an image query (leftmost top image) is submitted. They clearly show that image retrieval effectiveness is the highest when  $E$  is integrated into the generalized Dirichlet mixture and the lowest when the Euclidean distance is used. Indeed, the number of false alarms and false dismissals is the smallest for  $E + Dirichlet$  followed by  $E + Gibbs$  (exponentiation), followed by  $E$ , followed by  $DS^*$ , and so on. The worst similarity ranking is provided by the Euclidean distance.

Table 1 confirms our preceding observations about the performance of  $E + \text{Dirichlet}$  against the effectiveness of the other similarity measures. The findings remain true in Table 2, except for Faloutsos's measure.

## 6 Conclusion

In this paper, we have defined two distances: the dissimilitude distance  $DS^*$  and the similarity distance  $E$  and proposed three variants of a new retrieval effectiveness measure. When incorporated into the Gibbs random field and particularly to the generalized Dirichlet mixture, the distance  $E$  appears to be a good similarity measure. Empirical analysis of six similarity measures is conducted on color histograms of an image database and shows that retrieval effectiveness is the highest for  $E + \text{Dirichlet}$  and the lowest for the Euclidean distance.

Our current activities concern the design of new algorithms for color layout extraction (including spatial relationships identification) and image segmentation in order to get a more discriminating power in image retrieval and hence increase our system retrieval effectiveness.

**Acknowledgments.** We are grateful to VRQ and Canadian Heritage for their financial support.

## References

1. B. Bouguila, D. Ziou, and J. Vaillancourt. Maximum likelihood estimation of the generalized dirichlet mixture. *Tech. Rep., Dep. CS, Université de Sherbrooke*, 2002.
2. C. Faloutsos, W. Equitz, M. Flickner, W. Niblack, D. Petkovic, and R. Barber. Efficient and effective querying by image content. *IBM Research Center*, 1994.
3. S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. Pattern Anal. Much. Intell.*, 6, no. 6:721–741, November 1984.
4. B. Hill, Th. Roger, and F.W. Vorhagen. Comparative analysis of the quantization of color spaces on the basis of the cielab color-difference formula. *ACM Trans. on Graphics*, 16:109–154, April 1997.
5. R. Missaoui, M. Sarifuddin, J. Vaillancourt, Y. Hamouda, and H. Laggoun. A framework for image mining and retrieval. In *Proceedings of the SPIE Visual Communications and Image Processing, VCIP-03, Lugano, Switzerland*, pages 430–438, 2003.
6. B. Rémillard and C. Beaudoin. Statistical comparison of images using gibbs random fields. In *Proceedings of Vision Interface'99*, pages 612–617, 1999.
7. M. Swain and D. Baladar. Color indexing. *Computer Vision*, 7, No. 1:11–32, 1991.
8. M.P. Wand. *Data-based choice of histogram bin width*, volume Working paper series of Australian graduate school of management. University of New South Wales, Australia, 1996.

# Multimedia Retrieval Using Multiple Examples

Thijs Westerveld and Arjen P. de Vries

CWI, INS1, PO Box 94079, 1090 GB Amsterdam, The Netherlands

**Abstract.** This paper presents a variant of our generative probabilistic multimedia retrieval model. Evaluation on the TRECVID 2003 collection shows the new variant, a document generation approach, is suitable for information needs with multiple examples. Moreover, in combination with textual information, the new variant outperforms the original one.

## 1 Introduction

A commonly used paradigm in image and video retrieval is that of querying by example (QBE). An example document (image or video) is presented to the search engine, and similar documents are requested. A slightly modified form of this paradigm is adopted in the TRECVID video retrieval benchmarking effort [1]. An information request is called a *topic*. It consists of a textual description of the multimedia need accompanied by one or more image and/or video examples. The goal is to return a ranked list of shots that meet the information need.

Combining multiple visual examples to return one set (or ranked list) of similar documents can be problematic. Consider for example the topic shown in Figure 1. Here the information need is for shots of points being scored in basketball. The need is clarified by 6 different examples, some of them close-ups of the ball going through the basket, others showing overview shots of the playing court. No document will be highly similar to *all* examples. Clearly, we are looking for some sort of *OR*-functionality here; a query result should be similar to any of the examples, but not necessarily to all.

A common approach to handling multiple queries is to run separate queries for each example and combine the results afterwards. In such an approach, the final score for a document is a function of either the *scores* or the *ranks* for the individual examples [2,3,4]. It is however far from trivial to choose a combination function that works well for a variety of queries.



**Fig. 1.** Topic 101: ‘Find shots of a basket being made’.

The present work leaves this approach and captures all the different facets of a set of query examples in a single *topic model*. For retrieval, all documents in a

collection are compared to this single topic model and ranked accordingly. The rest of the paper is organised as follows. Section 2 describes a generative probabilistic approach to information retrieval. Section 3 discusses how this approach can be applied to image and video retrieval. Section 4 shows experimental results and Section 5 summarises our main conclusions.

## 2 Generative Probabilistic Retrieval

Following Sparck Jones et al. [5], and Lafferty and Zhai [6], we introduce random variables  $D$  and  $Q$  to represent a document and a query, and an event  $r$  to represent ‘relevant’, and try to answer the following “Basic Question”: *What is the probability that this document is relevant to this query?* This probability of relevance,  $P(r|D, Q)$ , can be estimated indirectly using Bayes’ rule:  $P(r|D, Q) = P(D, Q|r)P(r)/P(D, Q)$ . For ranking documents, we may avoid estimation of  $P(D, Q)$  using the odds of relevance:

$$\frac{P(r|D, Q)}{P(\bar{r}|D, Q)} = \frac{P(D, Q|r)P(r)}{P(D, Q|\bar{r})P(\bar{r})}, \quad (1)$$

where  $\bar{r}$  means not  $r$ . In the following,  $Q$  and  $D$  are assumed independent in the unrelevant case ( $\bar{r}$ ).

**Assumption 1.**  $P(Q, D|\bar{r}) = P(Q|\bar{r})P(D|\bar{r})$

Factoring the conditional probability  $P(D, Q|r)$  in different ways leads to two distinct, though probabilistically equivalent, models [6]. One model corresponds to *query generation*, and the other to *document generation*.

The query generation model results from factoring  $P(D, Q|r)$  as  $P(D, Q|r) = P(Q|D, r)P(D|r)$ , giving the following odds of relevance:

$$\frac{P(r|D, Q)}{P(\bar{r}|D, Q)} = \frac{P(D, Q|r)P(r)}{P(D, Q|\bar{r})P(\bar{r})} = P(Q|D, r) \cdot \underbrace{\frac{P(D|r)}{P(D|\bar{r})}}_{\text{prior odds}} \cdot \underbrace{\frac{P(r)}{P(Q|\bar{r})P(\bar{r})}}_{\text{independent of } D} \quad (2)$$

Since the goal is to rank documents, we can ignore the document independent terms. Also, we assume equal priors, i.e., a priori all documents are equally likely. This results in the following retrieval status value (RSV) for a document  $D$ :

$$\text{RSV}(D) = P(Q|D, r) \quad (3)$$

The document generation approach results from factoring  $P(D, Q|r)$  as  $P(D, Q|r) = P(D|Q, r)P(Q|r)$ , arriving at a different equation for the odds of relevance:

$$\frac{P(r|D, Q)}{P(\bar{r}|D, Q)} = \frac{P(D, Q|r)P(r)}{P(D, Q|\bar{r})P(\bar{r})} = \frac{P(D|Q, r)}{P(D|\bar{r})} \cdot \underbrace{\frac{P(Q|r)P(r)}{P(Q|\bar{r})P(\bar{r})}}_{\text{independent of } D} \quad (4)$$

Ignoring all factors independent of  $D$  for ranking gives the following RSV:

$$\text{RSV}(D) = \frac{P(D|Q, r)}{P(D|\bar{r})} \quad (5)$$

### 3 Generative Multimedia Retrieval

The next step is to define how to estimate the probabilities  $P(Q|D, r)$ ,  $P(D|Q, r)$  and  $P(D|\bar{r})$ . Documents in our case are video shots and queries are either (sets of) images or shots. We choose to represent a shot by a representative keyframe, thus all queries and documents are images. A variant in which temporal aspects are incorporated is presented in [3]. We estimate the (conditional) probabilities of queries and documents, by building a statistical model for each image. Other Generative approaches for multimedia retrieval include [7,8].

#### 3.1 Gaussian Mixture Models

The model assumes that an image is the outcome of a random process that generates  $n$ -dimensional feature vectors  $\mathbf{x} = (x_1, \dots, x_n)$ , where each feature vector describes a small, square block of pixels. The retrieval framework itself is independent of the specificities of the features; we have used DCT coefficients and  $x$ - and  $y$ -coordinates to capture colour, texture and position of a pixel block. In the remainder, the term *sample* is used to refer to both the feature vectors and the pixel blocks they describe. One or more images are represented as a bag of samples  $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{N_S}\}$ .

The samples are assumed to be generated by a mixture of Gaussian sources, where the number of Gaussian components  $N_C$  is fixed for all images in the collection. The Gaussian mixture model (GMM) is fully described by a set of parameters  $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_{N_C})$  defining the different components. Each component  $C_i$  is described by its prior probability  $P(C_i)$ , the mean  $\boldsymbol{\mu}_i$  and the variance  $\boldsymbol{\Sigma}_i$ , thus  $\boldsymbol{\theta}_i = (P(C_i), \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ . Details about estimating these parameters are deferred to Section 3.2. Equation 6 defines the probability of drawing one sample  $\mathbf{x}$  from a GMM with parameters  $\boldsymbol{\theta}$ .

$$p(\mathbf{x}|\boldsymbol{\theta}) = \sum_{i=1}^{N_C} P(C_i) \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma}_i|}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)} \quad (6)$$

The probability of drawing a bag of samples is simply the joint probability of drawing the individual samples:

$$p(\mathcal{X}|\boldsymbol{\theta}) = \prod_{i=1}^{N_S} p(\mathbf{x}_i|\boldsymbol{\theta}) \quad (7)$$

#### 3.2 Parameter Estimation

One way to look at mixture modelling for images is by assuming an image can show only so many different things, each of which is modelled by a Gaussian distribution. Each sample in a document is then assumed to be generated from one of these Gaussian components. This viewpoint, where ultimately each sample is explained by one and only one component, is useful when estimating the

GMM parameters. The assignments of samples  $\mathbf{x}_j$  to components  $C_i$  can be viewed as hidden variables, so the Expectation Maximisation (EM) algorithm [9] can be used. This algorithm iterates between estimating the a posteriori class probabilities for each sample (the E-step) given the current model settings, and re-estimating the components parameters based on the sample distribution and the current sample assignments (the M-step):

**E-step:** Estimate the hidden assignments  $h_{ij}$  of samples  $x_j$  to components  $C_i$ , for all samples and components.

$$h_{ij} = P(C_i|\mathbf{x}_j) = \frac{p(\mathbf{x}_j|C_i)P(C_i)}{\sum_{c=1}^{N_C} p(\mathbf{x}_j|C_c)P(C_c)} \quad (8)$$

**M-step:** Update the component's parameters to maximise the joint probability of component assignments and samples.  $\boldsymbol{\theta}^{\text{new}} = \arg \max_{\boldsymbol{\theta}} p(\mathcal{X}, \mathbf{H}|\boldsymbol{\theta})$ , where  $\mathbf{H}$  is the matrix with all sample assignments  $h_{ij}$ . More specifically:

$$\boldsymbol{\mu}_i^{\text{new}} = \frac{\sum_j h_{ij} \mathbf{x}_j}{\sum_j h_{ij}}, \quad (9)$$

$$\boldsymbol{\Sigma}_i^{\text{new}} = \frac{\sum_j h_{ij} (\mathbf{x}_j - \boldsymbol{\mu}_i^{\text{new}})(\mathbf{x}_j - \boldsymbol{\mu}_i^{\text{new}})^T}{\sum_j h_{ij}}, \quad (10)$$

$$P(C_i)^{\text{new}} = \frac{1}{N} \sum_j h_{ij} \quad (11)$$

The algorithm is guaranteed to converge to a local optimum. In previous experiments we found EM initialisation hardly influences the retrieval results [10].

### 3.3 Smoothing

Typicalities are more interesting than commonalities. Smoothing is a technique for explaining the common query terms, to reduce their influence on the ranking [11]. The estimates of the GMM are smoothed using interpolation with a general, background distribution – this technique is known as Jelinek-Mercer smoothing [12]. The smoothed version of the likelihood for a single sample  $\mathbf{x}$  becomes (cf. Equation 6):

$$p_{\text{smooth}}(\mathbf{x}|\boldsymbol{\theta}) = \kappa \left[ \sum_{i=1}^{N_C} P(C_i) \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma}_i|}} e^{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)} \right] + (1 - \kappa) p(\mathbf{x}), \quad (12)$$

where  $\kappa$  is a mixture parameter that can be estimated on training data with known relevant documents. The background density  $p(\mathbf{x})$  is estimated by marginalisation over all document models in a reference collection  $\mathcal{D}$ :

$$p(\mathbf{x}) = \sum_{d \in \mathcal{D}} p(\mathbf{x}|\boldsymbol{\theta}_d)P(d) \quad (13)$$

The reference collection  $\mathcal{D}$  can be the current collection, a representative sample of that, or, another *comparable* collection.

### 3.4 GMMs and the Retrieval Framework

In the GMM approach, each document  $D$  has 2 representations: a set of samples  $\mathcal{X}_D$  and a Gaussian mixture model  $\theta_D$  (the same holds for queries  $Q$ ). To relate this to the conditional probabilities introduced in Section 2, we estimate  $P(A|B, r)$  as the probability that the model of  $B$  ( $\theta_B$ ) generates the samples of  $A$  ( $\mathcal{X}_A$ ). Furthermore, to estimate  $P(A|\bar{r})$  we use the joint background density of all samples of  $\mathcal{X}_A$  (cf. Equation 13). Thus, the retrieval status values for query generation (Eq. 3) and document generation (Eq. 5) are estimated as

$$\text{RSV}_{\text{Qgen}}(D) = P(Q|D, r) \equiv P(\mathcal{X}_Q|\theta_D) \quad (14)$$

$$\text{RSV}_{\text{Dgen}}(D) = \frac{P(D|Q, r)}{P(D|\bar{r})} \equiv \frac{P(\mathcal{X}_D|\theta_Q)}{P(\mathcal{X}_D)} \quad (15)$$

## 4 Experiments

We evaluated the query and document generation variant of the generative probabilistic retrieval framework on the TRECVID 2003 search task [1]. For each document in the collection, and for each set of query examples, we build an 8-component GMM as described in Section 3.2 ( $N_C = 8$ ). Since we are interested in multiple-example queries, we regard samples from all available query images as a single set of query samples. We study two variants to represent the sets of query samples  $\mathcal{X}_Q$ . The first variant uses all available query samples, the second only those samples occurring in manually selected, *interesting* regions.<sup>1</sup> The same sets of samples are used to build topic models  $\theta_Q$  for the document generation approach.

### 4.1 Results

We have two model variants (query generation and document generation), and two ways of building query sample sets (full and regions). This amounts to four different system variants. Each of these is evaluated in isolation, as well as in combination with textual information. In the multimodal runs, we use a separate textual model, similar to the query generation approach described before.<sup>2</sup> For each shot a textual model is built from speech transcripts associated with the shot.<sup>3</sup> Assuming independence between the modalities, visual and textual models are used separately, and scores are combined afterwards. For details see [14]. Table 1 shows results for different experimental settings (the last column is explained in Section 4.2). Using full example images, query generation outperforms document generation, but if we select regions, the situation is reversed.

<sup>1</sup> Our manually selected query regions are available from <http://www.cwi.nl/projects/trecvid/trecvid2003/>.

<sup>2</sup> A document generation approach for the textual part is problematic, since the short text queries provide insufficient data to estimate proper topic models from.

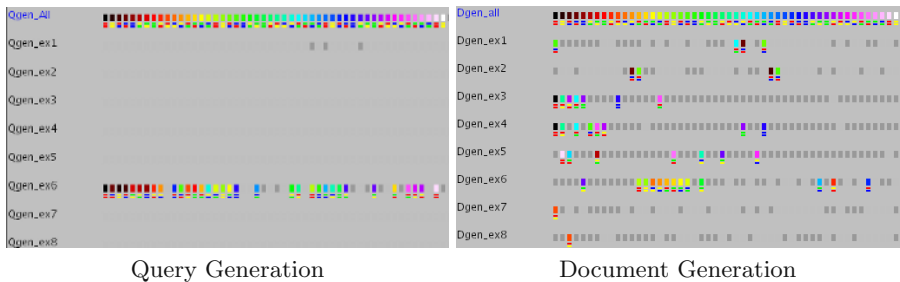
<sup>3</sup> The speech transcripts have been kindly provided by LIMSI [13].



Looking at the average precision scores per topic, rather than only at the mean, and inspecting the returned ranked lists for the different models, interesting differences are found. The query generation approach seems to be good at finding (near) exact matches, and is successful mainly when the set of examples is homogeneous (e.g. highly similar CNN baseball shots, or Dow Jones graphics). When a set of examples is less homogeneous, often a single example dominates the query generation results. Figure 2 shows this effect. In the document generation approach, the topic models seem to have learned important common aspects of the query examples, thus all examples contribute to the combined result (see Figure 2), and more generic matches are found. The fact that common aspects are learned, could be an explanation why selecting regions helps here. When a user indicates important regions, the topic models will be more focused and retrieve better documents. In the query generation approach, selecting regions does not help, since exact matching relies heavily on background similarity.

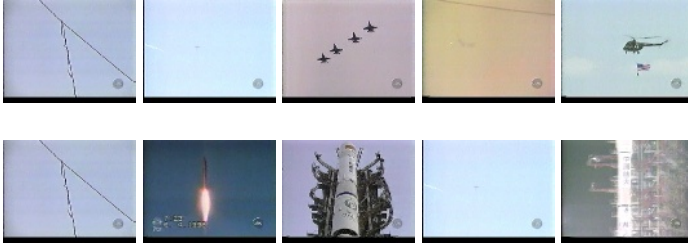
**Table 1.** Mean average precision for visual information only, and for a combination with text (MM) (text only: .130). Signs between brackets indicate a significant in/decrease of Dgen in comparison to the corresponding Qgen variant; Wilcoxon signed rank test, at confidence levels of 95% (+/-) and 99% (++)/-).

Qsamples	Qgen		Dgen		Dgen-BG	
	visual	MM	visual	MM	visual	MM
full	.028	.143	.026	.119 (-)	.034 (++)	.162 (+)
region	.026	.142	.026 (+)	.167	.034 (++)	.172 (++)



**Fig. 2.** Visualisations of the top 50s for the rocket launch query (topic0107). Each row represents retrieved documents for one run. Documents that are within the top 50 for the multiple-example run (top rows), are assigned a colour code, documents within the top 100 for this run are represented as grey rectangles. If a document from the multiple example run appears in another result, it is represented the same. Documents not in the top 100 for the multiple example run are not represented anywhere. Plots created using NIST's BeadPlot tool (see <http://www.itl.nist.gov/iaui/894.02/projects/beadplot/>).

In combination with textual information, the region based document generation approach is better than any query generation variant. The lower performance of the query generation approaches can be explained because the near-exact matches on visual content interfere with the textual ranking. In the document generation approach however, the visual information seems to provide the generic visual context, while the textual information zooms in on specific results. For example, for topics that ask for airplanes, helicopters or rocket launches, the visual model captures the fact that we are looking for an object against a background of sky. The textual information can then help to distinguish between specific objects. Figure 3 shows an example.



**Fig. 3.** Document generation results (top 5) for Rocket launch query (topic107). The visual information sets the context (top row, sky background) adding textual information fills in specifics (bottom row, rockets)

## 4.2 Automatically Selecting Regions

It is clear that selecting regions is useful for the document generation approach. Rather than selecting these manually, it is possible to automatically select important parts of an example image. The main idea is to select those parts of the example that differ most from the average image. Samples that are likely to be generated by any model should not influence the training process too much. A similar approach for text retrieval is studied in [15].

This can be achieved by incorporating background probabilities (Equation 13) in the training process. Again, hidden variables  $h_{ij}$  indicate the assignment of samples  $\mathbf{x}_j$  to components  $C_i$ , but now samples can also be assigned to the background, indicated by  $h_{BGj}$ . The EM-algorithm can be applied as before. The E-step changes to:

$$h_{ij} = P(C_i|\mathbf{x}_j) = \frac{p(\mathbf{x}_j|C_i)P(C_i)}{\sum_{c=1}^{N_C} p(\mathbf{x}_j|C_c)P(C_c) + p(\mathbf{x}_j)P(BG)} \quad (16)$$

$$h_{BGj} = P(BG|\mathbf{x}_j) = \frac{p(\mathbf{x}_j)P(BG)}{\sum_{c=1}^{N_C} p(\mathbf{x}_j|C_c)P(C_c) + p(\mathbf{x}_j)P(BG)}, \quad (17)$$

where  $P(BG|\mathbf{x}_j)$  is the posterior probability that  $\mathbf{x}_j$  is from the background, and  $P(BG)$  is the prior probability that we see background samples from the current model.

The M-step, does not update the background model  $p(\mathbf{x})$ . All we update are the component parameters (like in Equations 9,10, and 11), and the background prior ( $P(BG)$ ) for the current model.

$$P(BG)^{\text{new}} = \frac{1}{N} \sum_j h_{BGj} \quad (18)$$

Since common samples will be assigned to the background, only *distinguishing* samples are used in estimating the components' parameters. Figure 4 shows an example image and the regions that are automatically selected to build the model from.



**Fig. 4.** Sphinx example. Original image (left) and samples selected by EM algorithm (right).

The rightmost column of Table 1 shows the results for the new EM variant. A small (1%) sample from a comparable collection (the TRECVID 2003 development set) was used to estimate the background probabilities ( $P(\mathbf{x}_j)$ ) for the query samples. Clearly, using background probabilities during training helps. Automatically selecting regions using the new EM variant is almost as good as manually selecting important regions. Automatically finding distinguishing parts within manually selected regions gives another improvement.

## 5 Conclusions

This work presented two ways of applying generative probabilistic retrieval models to the problem of video retrieval: a query generation approach and a document generation approach. We showed that the query generation approach is not good at handling multiple-example queries. Usually, there is no document model in the collection that is likely to generate all available visual examples. In such cases, the query generation approach results in a model that explains, only one of the examples very well.

The document generation approach on the other hand, has to capture all information available in the examples in a limited number of Gaussian components. Therefore, it captures mainly things that are present in all examples, and thus builds a model that describes the commonalities shared by the examples. This leads to results that take all different examples into account. Often the things captured in the query models are of a generic, context-like nature (e.g., sky, grass, water). This turns out to be very useful in combination with textual information, where the results are far better than anything obtained so far using

the query generation approach. We showed also, specifically for the document generation approach, that indicating important regions in example images is useful for retrieval. Our automatic approach yields results comparable to manual region selection. Automatically selecting important parts within manually created regions gives another improvement (though slight) on the scores over using the user's manual selection as is.

Future work on the document generation model should prove whether the results would be more like exact matches when multiple-example queries are modelled by more components. Another plan is to investigate automatic selection of samples for the query generation approach.

## References

1. Smeaton, A.F., Kraaij, W., Over, P.: TRECVID 2003 - an introduction. In: TRECVID 2003 Workshop. (2003)
2. Jin, X., French, J.C.: Improving image retrieval effectiveness via multiple queries. In: Proceedings of the first ACM int. workshop on Multimedia databases. (2003)
3. Westerveld, T., Ianeva, T., Boldareva, L., de Vries, A.P., Hiemstra, D.: Combining information sources for video retrieval. In: TRECVID 2003 Workshop. (2003)
4. Natsev, A., Smith, J.R.: Active selection for multi-example querying by content. In: IEEE International Conference on Multimedia and Expo (ICME). (2003)
5. Sparck Jones, K., Walker, W., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments. *IP&M* **36** (2000)
6. Lafferty, J., Zhai, C.: Probabilistic IR models based on document and query generation. In: Language Modeling for Information Retrieval. (2003)
7. Blei, D.M., Jordan, M.I.: Modeling annotated data. In: ACM SIGIR 2003. (2003)
8. Jeon, J., Lavrenko, V., Manmatha, R.: Automatic image annotation and retrieval using cross-media relevance models. In: ACM SIGIR 2003. (2003)
9. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. *Journ. of the Royal Statistical Society, series B* **39** (1977)
10. Westerveld, T., de Vries, A.P.: Experimental result analysis for a generative probabilistic image retrieval model. In: ACM SIGIR 2003. (2003)
11. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: ACM SIGIR 2001. (2001)
12. Jelinek, F., Mercer, R.: Interpolated estimation of markov source parameters from sparse data. In: Proc. of the Workshop on Pattern Recognition in Practice. (1980)
13. Gauvain, J., Lamel, L., Adda, G.: The LIMSI broadcast news transcription system. *Speech Communication* **37** (2002)
14. Westerveld, T., de Vries, A.P., van Ballegooij, A., de Jong, F.M.G., Hiemstra, D.: A probabilistic multimedia retrieval model and its evaluation. *EURASIP Journal on Applied Signal Processing* **2003** (2003)
15. Hiemstra, D., Robertson, S., Zaragoza, H.: Parsimonious language models for information retrieval. In: ACM SIGIR 2004. (2004)

# A Discussion of Nonlinear Variants of Biased Discriminants for Interactive Image Retrieval

Xiang Sean Zhou<sup>1</sup>, Ashutosh Garg<sup>2</sup>, and Thomas S. Huang<sup>3</sup>

<sup>1</sup> Siemens Corporate Research, Princeton, NJ 08540, USA

<sup>2</sup> Google Inc., Mountain View, CA 94043, USA

<sup>3</sup> University of Illinois at Urbana Champaign, Urbana, IL 61801, USA

**Abstract.** During an interactive image retrieval process with relevance feedback, kernel-based or boosted learning algorithms can provide superior nonlinear modeling capability. In this paper, we discuss such nonlinear extensions for biased discriminants, or BiasMap [1,2]. Kernel partial alignment is proposed as the criterion for kernel selection. The associated analysis also provides a gauge on relative class scatters, which can guide an asymmetric learner, such as BiasMap, toward better class modeling. We also propose two boosted versions of BiasMap. Unlike existing approach that boosts feature components or vectors to form a composite *classifier*, our scheme boosts linear BiasMap toward a nonlinear *ranker* which is more suited for small-sample learning during interactive image retrieval. Experiments on heterogeneous image database retrieval as well as small sample face retrieval are used for performance evaluations.

## 1 Introduction

Within the last decade, numerous relevance feedback algorithms have been proposed for learning during interactive image retrieval [3,4,5,6,7,8]. For recent surveys see [9, 2]. Most early relevance feedback algorithms assumed Gaussian distribution for image classes. To relax this restrictive assumption, recent developments use either kernel machines or boosting methods to capture nonlinearities. Examples include kernel-based support vector machines [10,11,12,13], adaptive quasicomformal kernel (AQK) method [14], kernel-based BiasMap [1,2], Boosting approaches including FBoost or VBoost [4, 15], or a constrained similarity measures using SVM or AdaBoosting [16].

However, None of above addresses kernel selection in a principled way. In this paper, we propose kernel partial alignment for measuring kernel “fitness” which is especially suited for small-sample learning during image retrieval. It also provides a gauge on relative class scatters, which can guide a biased learner, such as BiasMap, toward better class modeling.

We also propose two boosting algorithms as alternatives to the kernel-based version for nonlinear distributions. Unlike existing approach that boosts individual feature components or vectors to form a composite and symmetric *classifier* [4,15], our scheme boosts multiple linear rankers (each of which uses all the available features) toward a nonlinear *ranker* which is more suited for information retrieval tasks [17].

It should be noted, however, that in content-based image retrieval (CBIR), the success of the learning module is contingent upon the effectiveness of the content representation

module. To focus on the issue of learning, we assume that a subset of the visual features, with some transformations, are relevant and informative. Although we do expect the features to be superfluous for any given retrieval task.

We will first briefly revisit the biased discriminant analysis (BDA) algorithm, upon which we will test our proposed kernel and boosting techniques. We use the term *BiasMap* as a shorthand for the class of algorithms based on linear or kernel BDA.

## 2 BiasMap Revisited

Given a set of positive and negative examples fed-back by the user, the aim is to find a ranking function from which we want not only that positive examples be ranked higher than the negative ones, but also that the positive examples be among top  $\kappa$  returns, with a minimal  $\kappa$ . In [1], the assumption was that the user is only interested in one class out of an unknown number of many. Thus the negative examples can come from an uncertain number ( $p$ ) of classes. If we could get sufficient training examples there is no need to distinguish so-called  $(1+p)$ -class learning from traditional two-class learning. However, when the training sample is small, such a treatment becomes useful.

### 2.1 Biased Discriminant Analysis (BDA)

An optimal discriminative transform matrix  $W_{opt}$  was formulated as follows :

$$W_{opt} = \arg \max \frac{|W^T S_y W|}{|W^T S_x W|} \quad (1)$$

where  $S_x$  and  $S_y$  are the scatter matrix estimates:

$$S_x = \sum_{i=1}^{N_x} (x_i - m_x)(x_i - m_x)^T, \quad S_y = \sum_{i=1}^{N_y} (y_i - m_x)(y_i - m_x)^T \quad (2)$$

$\{x_i, i = 1, \dots, N_x\}$  denote the positive examples, and  $\{y_i, i = 1, \dots, N_y\}$  denote the negative examples. Each element of these sets is a vector of length  $n$ , which is the dimension of the feature space.  $m_x$  is the mean vectors of the sets  $\{x_i\}$ . The term “biased discriminant analysis” shall not be confused with that of [18] where only the *statistical bias* of the original discriminant analysis was studied; while [1] also addresses statistical bias, it further explicitly models *class asymmetry* stemmed from the *subjective bias* of the user.

### 2.2 Kernel BDA (KBDA)

The rationale of kernel BDA, or of kernel machines in general, was to apply the original linear algorithm in a *feature space*,  $\mathcal{F}$ , which is related to the original space by a non-linear mapping

$$\phi : \mathcal{C} \rightarrow \mathcal{F} \quad | \quad x \rightarrow \phi(x) \quad (3)$$

where  $\mathcal{C}$  is a compact subset of  $\mathbb{R}^n$ , such that linearly non-separable configurations becomes separable in  $\mathcal{F}$ . However, this mapping can be formidably expensive thus will not be carried out explicitly, but through the evaluation of a kernel matrix  $K$  with components  $k(x_i, x_j) = \phi^T(x_i)\phi(x_j)$ . This is the same idea adopted by nonlinear support vector machines [19], kernel PCA, and kernel discriminant analysis [20,21].

In [1], it was shown that the optimal nonlinear feature space mapping is achieved through a weighted summation of kernel distances to the training points. The weights are the solutions to a generalized eigenanalysis problem. The retrieval process is then a simple search of nearest neighbors in this transformed space. However, the issue of kernel or kernel parameter selection was not studied in [1]. This is our next topic.

### 3 Kernel Partial Alignment

In this section, we introduce a measure called *kernel partial alignment* for kernel or kernel parameter selection. The idea is an adaptation of the kernel alignment of [22] for BiasMap. The aim is to measure the “alignment” between a Gram matrix and an ideal target matrix, and use the score as a goodness measure of the kernel.

**Definition 1 (Kernel Alignment [22]).** *The alignment of a kernel  $k$  on a sample  $S$  with the target matrix  $ll^T$  is:*

$$A = \frac{\langle K, ll^T \rangle_F}{\|ll^T\|_F \|K\|_F} = \frac{\langle K, ll^T \rangle_F}{N \|K\|_F} \quad (4)$$

where  $l$  is the class label vector taking values from  $\{-1, 1\}$ ,  $K$  is the Gram matrix for  $S$  using  $k$ ; and the Frobenius matrix inner product ( $\|K\|_F$  is the associated matrix norm) is adopted:  $\langle P, Q \rangle_F = \sum_{i,j} P_{ij}Q_{ij}$ .

The kernel alignment  $A$  provides a measure for selecting/combining kernels.

Using similar notations we define kernel partial alignment as follows:

**Definition 2 (Kernel Partial Alignment).** *The partial alignment of kernel  $k$  with the ideal target matrix on a sample  $S = \{x_1, \dots, x_{N_x}, y_1, \dots, y_{N_y}\}$ , with  $x_i$ 's being the positive set, is:*

$$A_{x|y}^P = \frac{\langle [K_{xx}, K_{xy}], [L_{xx}, L_{xy}] \rangle_F}{\|[K_{xx}, K_{xy}]\|_F \|[L_{xx}, L_{xy}]\|_F} \quad (5)$$

where  $K_{xx}$  is the  $N_x$  by  $N_x$  kernel matrix with elements  $k(x_i, x_j)$ ,  $K_{xy}$  is the  $N_x$  by  $N_y$  kernel matrix with elements  $k(x_i, y_j)$ ;  $L_{xx}$  and  $L_{xy}$  are of the same size as  $K_{xx}$  and  $K_{yy}$ , and with elements 1 and  $-1$ , respectively. (In case the data is unbalanced, the  $-1$  shall be replaced by  $-N_x/N_y$ .)

Notice that  $A^P$  by definition is asymmetric with respect to  $x$  and  $y$ .  $A^P$  is in fact an alignment based on part of the overall Gram matrix  $K$  and part of the target matrix  $ll^T$ , hence the name. While the original definition of kernel alignment accumulates within-class similarities for both classes, the partial definition accumulates only the similarities

among the positive points. Therefore, kernel partial alignment puts an emphasis on the biased treatment toward the positive class, matching the spirit of the BiasMap algorithm. As pointed out in [2], such biased treatment is essential for information retrieval tasks where the user is interested only in the positive class, and the clustering of the negative class is neither of interest nor achievable without compromising the clustering of the positive class in the transformed space. With  $A^P$ , we can answer the following questions: how to choose the right kernel or kernel parameter(s) for KBDA? And in addition, how to measure class scatter or “compactness” in the space induced by a kernel?

### 3.1 Kernel Selection

Kernel function can be regarded as a similarity measure. In fact, the distance in the feature space for two input vectors can be written as:

$$d_{ij}^2 = \|\phi(x_i) - \phi(x_j)\|^2 = k(x_i, x_i) + k(x_j, x_j) - 2k(x_i, x_j) \quad (6)$$

Since we can assume  $k(x_i, x_i)$  to be 1 (otherwise we can normalize it),  $k(x_i, x_j)$  corresponds to a negative (squared) distance measure, or a similarity measure.

Intuitively, the alignment  $A^P$  is a normalized difference between the positive within-class similarity and the between-class similarity. Hence, a kernel that maximizes this quantity captures the positive class distribution in a discriminative way. It was shown that high alignment leads to better generalization, at least for a Parzen window estimator [22]. A similar connection can be established for partial alignment.

$A^P$  can be used in much the same fashion for the purpose of kernel selection. For example, to choose the best spread parameter  $\sigma$  for a Gaussian RBF kernel, we can simply choose the  $\sigma$  value that gives the maximal  $A^P$ . Or, combining two kernels that are not aligned to each other can yield a better aligned combination.

However, in the following we focus on a rather interesting property of  $A^P$  that is not shared by full alignment  $A$ , which can be used to analyze relative class scatters.

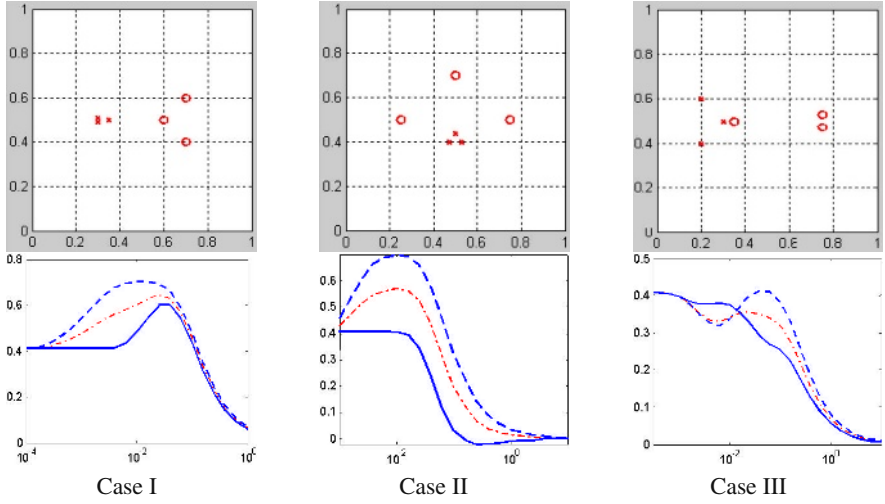
### 3.2 Relative Class Scatter Analysis

Full alignment  $A$  can be regarded as a measure for class “clusterization” for the two classes [22], but it cannot tell which class is more clustered. In the context of relevance feedback in image retrieval, we are interested in verifying that the underlying concept associated with the positive class is indeed supported by the visual features and the chosen kernel; or, the positive class is indeed less scattered than the negative class. Partial alignment  $A^P$  is an effective measure for *relative* class scatter analysis.

We use some synthetic data to illustrate the usefulness of  $A^P$ . Figure 1 shows three toy examples in 2-D space. For the figures on the second row, the horizontal axes represent the spread parameter  $\sigma$  for the Gaussian RBF kernel. The vertical axis represents alignments. One can make the following observations:

1. One class achieves a higher  $A^P$  than the other, if it is less scattered (case I, II);
2. Relative class scatter is scale dependent, e.g., for case III, at a very small scale both classes have three modes (one point per mode); when  $\sigma = 0.01$ , the “circle” class





**Fig. 1.** Partial alignment curves for comparing class scatters. The solid curves are partial alignments  $A^P$  for the “circle” classes, and the dashed curves represent the “cross” classes. The thinner dashed-dotted curves depict full alignment  $A$ .

becomes more clustered with two modes, while the cross class has three modes. But at  $\sigma = 0.1$ , the “cross” class merges into one mode, but the “circle” class still has two modes. Finally at very large scales, all classes have one mode; all points are similar to each other, so the curve converges to 0.

3. The higher the  $A^P$  curve reaches, the better is the learnability (e.g., by kernel Bi-asMap) of the positive class as opposed to the negative class; e.g., Case I and II the “cross” class is easy to learn than the “circle” class, whereas both classes are difficult in case III.
4. It can dip below 0, indicating a multi-mode distribution, possibly separated by the other class (Case II).

A simple case of reaching a negative  $A^P$  is to have two points per class,  $\{x_1, x_2\}$  and  $\{y_1, y_2\}$ , and  $x_1 = \alpha y_1 + (1 - \alpha)y_2$ , and  $x_2 = \beta y_1 + (1 - \beta)y_2$ , with  $0 < \alpha, \beta < 1$ . Then, with a (locally) concave kernel function (e.g., Gaussian RBF kernel), we have:

$$k(x_1, y_1) > \alpha k(y_1, y_1) + (1 - \alpha)k(y_2, y_1), \quad (7)$$

$$k(x_1, y_2) > \alpha k(y_1, y_2) + (1 - \alpha)k(y_2, y_2) \quad (8)$$

$$\Rightarrow k(x_1, y_1) + k(x_1, y_2) > k(y_1, y_1) + k(y_1, y_2) \quad (9)$$

Similarly, we have

$$k(x_2, y_1) + k(x_2, y_2) > k(y_2, y_1) + k(y_2, y_2) \quad (10)$$

These will lead to a negative  $A^P$  for class  $\{y_i\}$ .

Typically, a negative  $A^P$  indicates that a two-class assumption is really not appropriate, and one shall try splitting the modeling into three or more classes to achieve better

performance. In general, how to derive class-clustering structures from the *shape* of an  $A^P$  curve is an interesting future research direction.

Other measures exist for measuring relative class scatters in the feature space induced by a kernel. For example,  $W_{opt}$  of Eq. (4) is a measure of positive clusterization over negative-to-positive scatter. But this is in a reduced-dimensional space instead of the original space. It depends on the dimensionality reduction algorithm used.

To summarize, advantages of using kernel partial alignment  $A^P$  for measuring relative class scatters include: 1.  $A^P$  has values between  $-1$  and  $1$ ; 2. It is sharply centered around its expected value [22]; 3. It carries a simple geometrical interpretation—it is the inner product of two bi-dimensional vectors; 4. It is easy to compute.

### 3.3 Inverse Modeling During Retrieval

Using the relative class scatters, we can derive a new modeling paradigm. Assuming that the user is looking for image class  $\Theta$ , if based on the current training examples, we found out that the non- $\Theta$  class is more compact (i.e., has a higher  $A^P$  curve), what we shall do is to model the non- $\Theta$  class using kernel BiasMap, and return points that are far from non- $\Theta$  centroid in the transformed feature space. One example is looking for *non-face* images (an unusual scenario indeed), the right choice is to model the *non-non-face*, i.e., *face* class. Figure 2a shows the  $A^P$  curves for both classes.

A more practical example is searching for *man-made objects* versus *natural scenes*. If examples of natural scenes seem more clustered, finding a concise model for them may be better than modeling man-made object, and vice versa. (If they have similar scatters, with enough training example, BiasMap may not be the best choice any more as compared to typical symmetric classifiers such as Fisher discriminants or SVM.)

The modeling process can be dynamic—if the class under modeling becomes more scattered based on new examples, one can switch to its complementary class.

## 4 RankBoosted BiasMap

Another way to add nonlinear capability to BDA is to apply the boosting technique developed in the computational machine learning area [23]. The basic idea is to iteratively re-weight the training examples based on the outputs of some weak learners, with difficult-to-learn points receiving higher weights to enter the next iteration. The final learner is a weighted combination of the weak learners. RankBoost [17] is the boosting algorithm specifically designed for ranking functions, i.e., learners that assign ranks instead of a binary identifier to the training and testing data.

### 4.1 RankBoosted BiasMap

To apply RankBoost on the BiasMap algorithm, there can be several alternative implementations, depending upon the range to which the outputs of the BiasMap algorithm are mapped. To avoid hard thresholding, we choose the range of  $(0, 1)$ . The output of the BiasMap algorithm is expressed in ranks  $r(x)$ , with  $r(x) \in N$ , the set of natural numbers.  $r(x)$  is defined as the smaller the better. However the ranking function  $h(x)$  for

RankBoost [17] is the larger the better. Therefore  $r(x)$  needs to be transformed before used as the input to RankBoost. A sigmoid-like transformation is adopted:

$$h(x) = 1 - \frac{1}{1 + e^{-\tau(r(x)-\theta)}}, \quad (11)$$

As the weak learner, BiasMap can take weighted training examples in a natural way by using weighted scatter matrix estimates, which places the positive centroid nearer to the points of higher weights, and also emphasizes the scattering directions of them. The weighted covariance matrix plug-in estimate for a set of vectors  $\{x_i\}$  with weights  $\{w_i\}$ , assuming  $\sum_i w_i = 1$ , is

$$C = \sum_i \left( (x_i - \sum_i w_i x_i) w_i (x_i - \sum_i w_i x_i)^T \right). \quad (12)$$

**Algorithm: RankBoost with BiasMap as weak learner**

Given: positive set  $X_1$ , and negative set  $X_0$  (notations in accordance with [17]);

Initialize training example weights  $w_1(x) = 1/|X_i|$ , for  $x \in X_i, i = 0, 1$ ;

For  $t = 1, \dots, T$ :

- Train a weak BiasMap using weights  $W_t$ : use weighted covariance matrix as the scatter matrix estimates (Equation (12)). Outputs are in  $r_t(x)$ ;
- Get weak hypothesis  $h_t : x \rightarrow (0, 1)$ , using Equation (11);
- Set  $\alpha_t = \frac{1}{2} \ln(\frac{1+\varsigma_t}{1-\varsigma_t})$ , with  $\varsigma_t = \sum_{x_0, x_1} w_t(x_0) w_t(x_1) (h_t(x_1) - h_t(x_0))$ ;
- Update the weights:
  - a.  $w_{t+1}(x) = w_t(x) \exp((-1)^i \alpha_t h_t(x))$ , for  $x \in X_i, i = 0, 1$ ;
  - b. Normalize the weights to sum to 1 separately for positive and negative examples.

Output the final hypothesis:  $H(x) = \sum_{t=1}^T \alpha_t h_t(x)$ .

---

## 4.2 A Customized Version of RankBoost

The direct implementation of RankBoost shown above, although backed by justification in terms of error bound analysis, did not give the best empirical results for the problem at hand. This is not surprising because the bound is not exact [17]. We tested several ad-hoc implementations, and found out that some performs better and faster. Below we describe RankBoost.H, a heuristic-based formulation that is tuned toward specific system requirements and setup for image retrieval.

**Algorithm: RankBoost.H with BiasMap as weak learner**

... (Inputs and initialization steps same as above);

For  $t = 1, \dots, T$ :

- Train a weak BiasMap using weights  $W_t$ : Outputs are in  $r_t(x)$ ;
- Set  $\alpha_t = \max \{0, \sum_{x_1 \in X_1} [r_t(x_1) < \kappa] w_t(x_1) - \sum_{x_0 \in X_0} [r_t(x_0) < \kappa] w_t(x_0)\}$ , here the notation  $[\Xi] = 1$  when the predicate  $\Xi$  holds, and 0 otherwise.

– Update the weights:

$$\text{a. } w_{t+1}(x) = \begin{cases} w_t(x)r_t(x) & \text{if } x \in X_1 \\ w_t(x)\frac{1}{r_t(x)} & \text{if } x \in X_0 \end{cases}$$

b. Normalize the weights to sum to 1 separately for positive and negative examples.

Output the final hypothesis:  $H(x) = \sum_{t=1}^T \alpha_t \frac{1}{r_t(x)}$ .

Intuitively, RankBoost.H has a more aggressive re-weighting scheme and a performance measure that requires more positive and less negative examples in top  $\kappa$  returns, which reflects the scenario for an image retrieval system. Although both linear and kernel-based BiasMap can be used as the weak learner, the linear version shall suffice. Because of the dynamic weighting of different parts of the training set, boosting can be regarded as an indirect nonlinear extension for linear BiasMap.

It should be noted that the RankBoosting algorithms proposed here are in principle very different from that of [4], where a very large set of features (in the thousands or millions) was used and the boosting was applied on single-feature naive Bayesian classifiers. Here we use a smaller feature set (around 40 components for the Corel testing, and 256 for face and non-face testing), but more complicated weaker learners (i.e., subspace learning through BiasMaps, each of which makes use of the full feature set).

### 4.3 Kernel or Boosting?

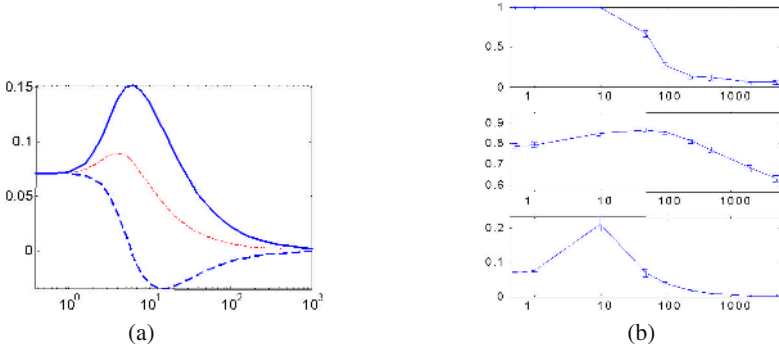
Kernel approach is powerful and flexible in that one can apply different kernels. However, the selection of kernel and kernel parameter is still an unsolved problem. Kernel alignment-based approach, although can narrow down the search space, may still overfit small training sample (as we shall see in the experiments). Boosting, on the other hand, is very fast and very simple to implement with minimal number of parameters to tune, and is believed to be less prone to overfitting. However, it is more restricted and provides less modeling freedom. In our experiments, the two perform comparably.

## 5 Experiments and Evaluations

In addition to a heterogeneous image database from the Corel dataset, we also use a face and non-face image set for the evaluation of the proposed algorithms. The reasons for using the face dataset is that the Corel set, though widely used, does not always provide convincing and fair results. The labeled classes are sometimes misleadingly described. E.g., “horses” are actually “horses in grassy background”—Oftentimes, it was the grass that kept the cluster together, not the horses; For the face and non-face problem, one can use pixel values as features, and this decouples the feature extraction and learning process, and allows for easier benchmarking of different learning algorithms.

### 5.1 Benchmark Testing on Retrieving Faces

The 1000 faces and 1000 non-face images are 16-by-16 gray scale and the original pixel values are used to form a 256-dimensional feature vector [2].



**Fig. 2.** (a). Partial alignment curves 100 face (solid) and 100 non-face (dashed) images. The thin dash-dotted curve is the full alignment  $A$ . (b). Comparing hit rates and partial alignments with varying kernel scales.

For evaluation, we use both precision and separation (i.e., discrimination) measures. The latter promotes the effort of pushing negative examples away from the positives. For precision we use *hit rate*. For separation we use *disagreement* and *rank difference*:

**Definition 3 (Disagreement).** *Disagreement* [17] is the fraction of pairs of positive and negative examples that are misordered:  $\text{disagreement} = \frac{1}{N_x N_y} \sum_{i,j} [r(x_i) > r(y_j)]$ .

**Definition 4 (Rank difference).** *Rank difference* is the averaged difference in rank for all positive and negative pairs:  $\text{rank\_difference} = \frac{1}{N_x N_y} \sum_{i,j} (r(y_j) - r(x_i))$ .

**Definition 5 (Hit rate).** *Hit rate in top  $k$*  is the averaged number of positive images within the top  $k$  returns.

**Kernel selection through maximum alignment.** The first experiment is to test kernel partial alignment using average performance of KBDA from multiple runs. Figure 2b shows the mean and standard deviation curves from 50 rounds of testing. Each round 100 face and 100 non-face images were randomly sampled to form the training set. The hit rate for the training set (in top 100) and the hit rate for all the 1000 faces (in top 1000) are shown as the top two figures, under varying values for the spread parameter of a Gaussian kernel. The bottom figure shows the corresponding partial alignments. It is observed that the maximum of the partial alignment curve aligns with the knee-point of the training curve (top), and is close to the maximum of the test hit rate curve (middle). The slight mis-alignment shows signs of over fitting. This is due to the relative small training set. (The maximum of full alignment is worse-see Figure 2a). The small standard deviation error bars on the curve seems to verify the theorem in [22] which says that the alignment is sharply centered on its expected value.

In the following, we use the maximum of  $A^P$  as guidance and perform a cross-validated fine search in the neighborhood for the best kernel parameter. This can provide a dramatic speedup (in some cases over 100 times) as compared to a global search.

**BDA vs. query movement (QM), whitening transform (WT), and SVM.** We first compare BDA with existing alternatives. Query movement, a popular technique in information retrieval, is implemented here by taking the positive centroid as the new query point and returning its nearest neighbors. Whitening transform uses the Mahalanobis distance instead of the Euclidean distance used by QM, which implies a PCA-like whitening transform of the representation space based on positive examples. For SVM we use a RBF kernel with a numerically sought optimal spread parameter. With 100 positive examples and a varying number of negative examples randomly selected as the training data, the results are shown in Figure [reffig:comparea](#). For BDA, the parameters are  $\mu = 0.01$  and  $\gamma = 0$  (See [2] for definitions). Unless otherwise noted, all results are averages of 20 rounds of independent testing. The averaged disagreements are: QM: 23%, WT: 0.4%, SVM: 0%, and BDA: 0.01%.

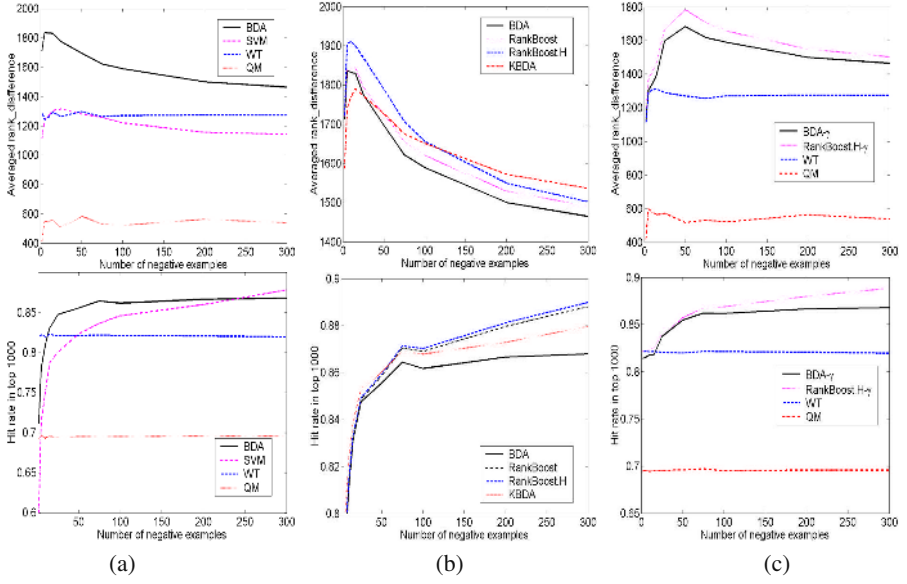
It is obvious that SVM, although can perfectly rank the positive and negative examples in the right order (disagreement = 0%), performs worse in terms of both rank difference and hit rate when the number of negative examples is small ( $< 200$ ). When the number of negatives is less than 10, BDA is below WT in terms of hit rate. This is due to the highly singular negative scatter matrix with extremely biased eigenanalysis results. In practice this means that even though BDA can push away the negative examples (high rank differences), this action, if not moderated, may hurt the effort of clustering positive examples. The trade-off between rank difference and hit rate can be achieved through “ $\gamma$ -adaptation”, which will be discussed subsequently.

**Nonlinear BiasMap using boosting and kernel.** The nonlinear variants of BDA are compared in a similar setting, with 20 boosting iterations and the same RBF kernel for KBDA as that of SVM above. For RankBoost, the parameters of Equation (11) are  $\tau = 0.01$ ,  $\theta = 1000$  (numerically sought to achieve the best performance for hit rate in top 1000). For RankBoost.H,  $\kappa = 100$ . The results are shown in Figure 3b.

We make the following observations: The nonlinear schemes, boosting and kernel approach, provide comparable improvement to BDA. (Since BDA is a strong performer itself, boosting yields consistent but not dramatic improvement [23].) In terms of averaged number of positives in top 100 returns, RankBoost has 67.6, while RankBoost.H gives 81.4. All of them reduce the averaged disagreement of BDA to 0.

**$\gamma$ -Adaptation.** When the number of negative examples  $N_y$  is very small (say,  $< 10$ ), the negative scatter matrix should also be shrunk toward an identity matrix to limit the statistical bias ([2], p. 124). Thus, it is necessary to adaptively adjust  $\gamma$  according to  $N_y$ , with a larger  $\gamma$  for a smaller  $N_y$ . An ad hoc but sensible choice is:  $\gamma = \exp(-N_y/\zeta)$ .

When  $N_y = 0$ ,  $\gamma = 1$ , BDA reduces to whitening transform. Figure 3c shows the new results with  $\zeta$  set to be 5. The disagreement scores are: QM: 22%, WT: 0.3%, BDA- $\gamma$ : 0.12%, and RankBoost.H- $\gamma$ : 0%. The suffix “- $\gamma$ ” denotes the  $\gamma$ -adapted version of the corresponding algorithm. Comparing with Figure 3a, we clearly see in Figure 3c, for small  $N_y$ ’s, the trade-off between hit rate and rank difference / disagreement. Indeed, the performance goals in terms of *discrimination* and *precision* can be traded off by adaptively adjusting the values of  $\zeta$  or  $\gamma$ , so that one can achieve a desired level of performance that is an “interpolation” of Figure 3a and c.



**Fig. 3.** Performance comparisons (for detailed discussions see text).

**Table 1.** Average hit rate in top 100 for 500 rounds of testing.

QM	WT ( $\mu = 0.1$ )	BDA ( $\mu = 0.1, \gamma=0$ )	RankBoost.H ( $\kappa=100$ )	KBDA (RBF: $\sigma=0.7$ )
62.7%	70.4%	74.2%	77.5%	79.1%

## 5.2 Image Database Testing

The second experiment is to apply the proposed algorithms for content-based image retrieval on a heterogeneous dataset. A fully labeled set of 500 images from Corel image set is used for testing. It consists of five classes, each with 100 images. Each image is represented by a 37 dimensional vector, which consists of 9 color moments, 10 texture features, and 18 edge-based structure features [2]. Each round 10 positive and 10 negative images are drawn as training samples. For each round the hit rate in the top 100 returns is recorded. 500 rounds of testing are performed on all 5 classes and the averaged hit rates are shown in Table 1. It is worth pointing out that based on the performance shown here, RankBoost.H and KBDA both outperform a one-class SVM based kernel density estimator (cf. [11]).

## 6 Conclusion

We presented a kernel partial alignment scheme for kernel parameter selection. Its “serendipitic” value toward relative class scatter analysis was also discussed. We also proposed new boosting algorithms with BiasMap rankers as the weak learner. These

algorithms are suited for the small-sample learning problem during relevance feedback and were shown to have better performance than off-the-shelf tools used in the literature. The proposed algorithms can also be used for other applications.

## References

1. Zhou, X.S., Huang, T.S.: Small sample learning during multimedia retrieval using biasmap. In: Proc. IEEE CVPR, Hawaii. (2001) 11–17
2. Zhou, X.S., Rui, Y., Huang, T.S.: Exploration of Visual Data. Kluwer Academic Publishers (2003)
3. Su, Z., Li, S., Zhang, H.: Extraction of feature subspaces for content-based retrieval using relevance feedback. In: ACM Multimedia. (2001) 98–106
4. Tieu, K., Viola, P.: Boosting image retrieval. (In: Proc. IEEE CVPR, South Carolina)
5. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. In: Proc. Int'l Conf. Machine Learning. (2000) 999–1006
6. Vasconcelos, N., Kunt, M.: Content-based retrieval from image databases: Current solutions and future directions. In: Proc. IEEE ICIP, Greece. (2001)
7. Wu, Y., Tian, Q., Huang, T.S.: Discriminant-EM algorithm with application to image retrieval. In: Proc. IEEE CVPR, South Carolina. (2000) 222–227
8. Dong, A., Bhanu, B.: Active concept learning for image retrieval in dynamic databases. In: Proc. ICCV. (2003)
9. Worring, M., Smeulders, A., Santini, S.: Interaction in content-based image retrieval: a state-of-the-art review. In: Int'l Conf. on Visual Info. Sys., Lyon, France. (2000)
10. Hong, P., Tian, Q., Huang, T.S.: Incorporate support vector machines to content-based image retrieval with relevant feedback. In: Proc. IEEE ICIP, Vancouver, Canada. (2000)
11. Chen, Y., Zhou, X.S., Huang, T.S.: One-class svm for learning in image retrieval. In: Proc. IEEE ICIP, Greece. (2001)
12. Tong, S., Chang, E.: Support vector machine active learning for image retrieval. In: Proc. ACM Multimedia, Ottawa, Canada. (2001)
13. Zhang, L., Lin, F., Zhang, B.: Support vector machine learning for image retrieval. In: Proc. IEEE ICIP, Greece. (2001)
14. Heisterkamp, D., Peng, J., Dai, H.: An adaptive quasiconformal kernel metric for image retrieval. In: Proc. IEEE CVPR, Hawaii. (2001) 388–393
15. Howe, N.R.: A closer look at boosted image retrieval. In: Proc. CIVR'03. (2003)
16. Guo, G.D., Jain, A.K., Ma, W.Y., Zhang, H.J.: Learning similarity measure for natural image retrieval with relevance feedback. IEEE Trans. Neural Networks **13** (2002) 811–820
17. Freund, Y., Iyer, R., Schapire, R.E., Singer, Y.: An efficient boosting algorithm for combining preferences. In: Int'l Conf. Machine Learning. (1998) 170–178
18. Dipillo, P.: Biased discriminant analysis: Evaluation of the optimum probability of classification. Commun. Statist.-Theor. Meth. **8** (1979) 1447–1457
19. Vapnik, V.: The nature of statistical learning theory. Springer, New York (1995)
20. Baudat, G., Anouar, F.: Generalized discriminant analysis using a kernel approach. Neural Computation **12** (2000) 2385–2404
21. Mika, S., Rätsch, G., Müller, K.R.: A mathematical programming approach to the kernel fisher algorithm. In: NIPS-13. (2001) 591–597
22. Cristianini, N., Shawe-Taylor, J., Elisseeff, A., Kandola, J.: On kernel-target alignment. In: NIPS. (2001)
23. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Proc. Int'l Conf. on Machine Learning. (1996) 148–156



# Salient Objects: Semantic Building Blocks for Image Concept Interpretation

Jianping Fan<sup>1</sup>, Yuli Gao<sup>1</sup>, Hangzai Luo<sup>1</sup>, and Guangyou Xu<sup>2</sup>

<sup>1</sup> Department of Computer Science, University of North Carolina  
Charlotte, NC 28223, USA

<sup>2</sup> Department of Computer Science, Tsinghua University  
Beijing, CHINA

**Abstract.** Interpreting semantic image concepts via their dominant compounds is a promising approach to achieve effective image retrieval via keywords. In this paper, a novel framework is proposed by using the salient objects as the semantic building blocks for image concept interpretation. This novel framework includes: (a) using machine learning technique to achieve automatic detection of the salient objects; (b) using Gaussian mixture model for semantic image concept interpretation by exploring the quantitative relationship between the semantic image concepts and their dominant compounds, i.e., salient objects. Our broad experiments on **natural images** have obtained significant improvements on semantic image classification.

## 1 Introduction

The success of most existing content-based image retrieval (CBIR) systems is often limited and largely depends on effectiveness of the underlying image patterns that are selected for image content representation and feature extraction [1]. Four approaches are widely used for image content representation:

(a) *Scene-based approaches* treat entire images as the underlying image patterns for feature extraction and only the global visual properties are used for image content representation [8-12]. The major advantage of the scene-based approaches is that no segmentation is involved, but they may not work well for images that consist of individual objects [2-4], especially when the individual objects are used by human beings to interpret the semantics of images.

(b) *Region-based* or *blob-based approaches* take homogeneous image regions or connected image regions with consistent color or texture (i.e., image blobs) as the underlying image patterns for feature extraction [5-8]. The major problem for such approaches is that homogeneous image regions and image blobs cannot capture enough information of image semantics and thus their visual features do not have the ability to discriminate among different semantic image concepts.

(c) *Object-based approaches* take semantic objects as the representative image patterns for feature extraction [13-15]. The major problem for such approaches is that automatic semantic object extraction in general is still difficult because homogeneous image regions in color or texture do not correspond to the semantic objects directly [2-4].

(d) *Annotation-based approaches* use manual text annotation for image content interpretation [16-18]. The strength of such approaches is that query concepts can be expressed directly in terms of keywords which are meaningful to human users, but manual text annotation is too expensive for a large number of images [19-20].

We believe that an effective approach for image content representation should be able to avoid manual text annotation and semantic object extraction, but it should be able to enhance the ability of low-level visual features to discriminate among different semantic image concepts. In order to achieve that, we need certain means to detect suitable image patterns automatically and relate such image patterns to the most relevant semantic image concepts quantitatively.

Based on this understanding, we propose a novel framework to achieve automatic interpretation of semantic image concepts by using the *salient objects*. This paper is organized as follows: Section 2 presents a novel image content representation framework by using the salient objects; Section 3 introduces an automatic technique for salient object detection; Section 4 introduces a semantic image concept interpretation framework by using *Gaussian mixture model*; Section 5 shows our broad experiments on natural images; We conclude in Section 6.

## 2 Image Content Representation via Salient Objects

As mentioned above, the quality of low-level visual features largely depends on the underlying image patterns that are selected for image content representation and feature extraction. The low-level visual features, that are extracted by using entire images or homogeneous image regions, do not have the ability on discriminating among different semantic image concepts because the homogeneous image regions cannot describe the semantics of an image effectively. On the other hand, extracting the visual features from semantic image objects is very hard because automatic semantic object extraction in general is still difficult [2-4]. Therefore, there has been great interest in developing more effective image content representation framework such that a middle-level understanding about image contents can be achieved.

In order to achieve a middle-level understanding about image contents, we propose a novel semantic-sensitive image content representation framework by using the salient objects. The salient objects are defined as the visually distinguishable image compounds [14] or the global visual properties of whole images that can be identified by using the spectrum templates in the frequency domain [10]. For example, the salient object “sky” is defined as the connected image regions with large sizes (i.e., dominant image regions) that are related to the human semantics “sky”. The salient objects that are related to the global visual properties in the frequency domain can be obtained easily by using wavelet transformation [10]. In the following discussion, we will focus on modeling and detecting the salient objects that are related to the visually distinguishable image compounds. In addition, the *basic vocabulary* of such salient objects can be obtained by using the taxonomy of image compounds of *natural scenes* as shown in Fig. 1.

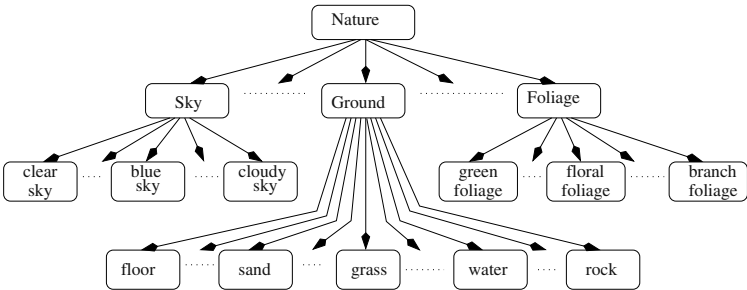


Fig. 1. Examples of the taxonomy of natural images.

3 Salient Object Detection

The objective of image analysis is to parse the natural images into the salient objects in the vocabulary under certain vision purposes. Thus each type of the salient objects can be taken as an abstraction of some properties decided by certain vision purposes. Based on the basic vocabulary as shown in Fig. 1, we have designed a set of detection functions and each function is able to detect one certain type of these salient objects in the basic vocabulary.



Fig. 2. The flowchart for automatic semantic salient object extraction.

We use our detection function for the salient object “beach sand” as an example to show how we can design our salient object detection functions. As shown in Fig. 2, image regions with homogeneous color or texture are first obtained by using mean shift techniques [22]. Since the visual characteristics of a certain type of salient objects will look differently at different lighting and capturing conditions [2-4], using only one image is very difficult to represent its characteristics and thus this automatic image segmentation procedure is performed on a set of training images with the salient object “beach sand”. The homogeneous regions obtained from the training images, that are related to the salient object “beach sand”, are selected and labeled by human interaction.

Region-based low-level visual features, such as 1-dimensional coverage ratio (i.e., density ratio between the pixels of real object region and the pixels of its rectangular box) for shape representation, 6-dimensional region locations (i.e., 2-dimensions for region center and 4-dimensions to indicate the rectangular box for coarse shape representation), 7-dimensional LUV dominant colors and color variances, 14-dimensional Tamura texture, and 28-dimensional wavelet texture features, are extracted for characterizing the visual properties of these labeled

image regions that are explicitly related to the salient object “sand field”. The 6-dimensional region locations will be used to determine the contextual relationships among different types of salient objects. The context relationships among different types of salient objects (i.e., coherence among different types of salient objects) can be used to distinguish some salient objects with similar visual properties such as “beach sand” and “road”, where the salient object “beach sand” has strong coherence with the salient object “sea water”.

We use *one-against-all* rule to label the training samples  $\Omega_{g_j} = \{X_l, G_j(X_l) | l = 1, \dots, N\}$ : positive samples for a specific visual salient object of “beach sand”  $G_j$  and negative samples. Each labeled training sample is a pair  $(X_l, G_j(X_l))$  that consists of a set of region-based low-level visual features  $X_l$  and the semantic label  $G_j(X_l)$  for the corresponding labeled homogeneous image region.

Based on the available visual features and labels, an image region classifier is learned from these labeled homogeneous image regions. We use Support Vector Machine (SVM) to learning the binary image region classifier [12]. Consider a binary classification problem with linearly separable sample set  $\Omega_{g_j} = \{X_l, G_j(X_l) | l = 1, \dots, N\}$ , where the semantic label  $G_j(X_l)$  for the labeled homogeneous image region with the visual feature  $X_l$  is either +1 or -1. For the positive samples  $X_l$  with  $G_j(X_l) = +1$ , there exists the transformation parameters  $A$  and  $b$  such that  $A \cdot X_l + b > +1$ . Similarly, for negative samples  $X_l$  with  $G_j(X_l) = -1$ , we have  $A \cdot X_l + b > -1$ . The margin between these two supporting planes will be  $2/\|A\|^2$ . SVM is then designed for maximizing the margin with the constraints  $A \cdot X_l + b > +1$  for the positive samples and  $A \cdot X_l + b > -1$  for the negative samples.

Our current implementation of SVM classifier is based on Gaussian function kernel. The margin maximization procedure is then transformed into:

$$\arg \min_{A, b, \xi} \frac{1}{2} A \cdot A + \sum_{l=1}^N \xi_l \quad (1)$$

$$G_j(A \cdot \Phi(X_l) + b) \geq 1 - \xi_l$$

where  $\xi_l \geq 0$  is the pre-defined error rate,  $C > 0$  is the penalty parameter of the error term,  $\Phi(X_l)$  is the function that maps  $X_l$  into higher dimensional space (i.e., feature dimensions plus the dimension of response) and the kernel function is defined as  $\kappa(X_i, X_j) = \Phi(X_i)^T \Phi(X_j)$ . In our current implementation, we select radial basis function (RBF),  $\kappa(X_i, X_j) = \exp(-\gamma \|X_i - X_j\|^2)$ ,  $\gamma > 0$ .

## 4 Interpretation of Semantic Image Concepts

In order to interpret the semantic image concept quantitatively by using the relevant salient objects, we have also proposed a novel image context integration framework via Gaussian mixture model. The class distribution for different types of the relevant salient objects is approximated by using a Gaussian mixture model with  $\kappa$  mixture Gaussian components:



**Fig. 3.** The detection results of salient objects: (a) elephant; (b) cat.

$$P(X, C_j, \kappa, \omega_{c_j}, \Theta_{c_j}) = \sum_{i=1}^{\kappa} P(X|S_i, \theta_{s_i}) \omega_{s_i} \quad (2)$$

where  $\kappa$  indicates the optimal number of mixture Gaussian components,  $\Theta_{c_j} = \{\theta_{s_i}, i = 1, \dots, \kappa\}$  is the set of the parameters for these mixture Gaussian components,  $\omega_{c_j} = \{\omega_{s_i}, i = 1, \dots, \kappa\}$  is the set of the relative weights among these mixture Gaussian components,  $X = (x_1, \dots, x_n)$  is the  $n$ -dimensional visual features that are used for representing the relevant salient objects. For example, the semantic concept, “beach scene”, is related to at least three types (classes) of the salient objects such as “sea water”, “sky pattern”, “beach sand” and other hidden image patterns.



**Fig. 4.** The detection results of salient objects: (a) grass; (b) sky.

The visual characteristics of a certain type of these salient objects will look differently at different lighting and capturing conditions. For example, the salient object “sky pattern”, it consists of various appearances, such as “blue sky pattern”, “white(clear) sky pattern”, “cloudy sky pattern”, and “sunset/sunrise sky pattern”, which have very different properties on color and texture under different viewing conditions. Thus, the data distribution for each type of these salient objects is approximated by using multiple mixture Gaussian components to keep the variability within the same class.



Fig. 5. The detection results of salient objects: (a) beach sand; (b) road.



Fig. 6. The detection results of salient objects: (a) sunset/sunrise; (b) snow.

For a certain semantic image concept, the optimal number of mixture Gaussian components and their relative weights are acquired automatically through a machine learning process by using our incremental EM algorithm [22]. Linear discriminant analysis is also performed on the original visual features to obtain a new transformed feature space, such that the independence among different types (classes) of the relevant salient objects and the discrimination among various semantic image concepts can be maximized [21].

5 Performance Evaluation

Our experiments are conducted on two image databases: photography database that is obtained from Google search engine and Corel image database. The photography database consists of 35000 digital pictures. The Corel image database includes more than 125,000 pictures with different image concepts.

Precision  $\rho$  and recall  $\varrho$  are used to measure the average performance of our detection functions:

$$\rho = \frac{\eta}{\eta + \varepsilon}, \qquad \varrho = \frac{\eta}{\eta + \vartheta} \tag{3}$$

where  $\eta$  is the set of true positive samples that are related to the corresponding type of salient object and detected correctly,  $\varepsilon$  is the set of true negative samples that are irrelevant to the corresponding type of salient object and detected incorrectly, and  $\vartheta$  is the set of false positive samples that are related to the





**Fig. 7.** The semantic image classification and annotation results for the concept “garden” with the most relevant salient objects.

corresponding type of salient object but mis-detected. The average performance for some detection functions is given in Table 1. Some results for our detection functions of salient objects are shown in Fig. 3, Fig. 4, Fig. 5 and Fig. 6.

The *benchmark metric* for classifier evaluation includes *classification precision*  $\alpha$  and *classification recall*  $\beta$ . They are defined as:

$$\alpha = \frac{\pi}{\pi + \tau}, \quad \beta = \frac{\pi}{\pi + \mu} \quad (4)$$

where  $\pi$  is the set of true positive samples that are related to the corresponding semantic concept and classified correctly,  $\tau$  is the set of true negative samples that are irrelevant to the corresponding semantic concept and classified incorrectly,  $\mu$  is the set of false positive samples that are related to the corresponding semantic concept but mis-classified.

As mentioned above, two key issues may affect the performance of the classifiers: (a) the performance of our detection functions of salient objects; (b) the performance of the underlying classifier training techniques. Thus the real impacts for semantic image classification come from these two key issues, the *average precision*  $\bar{\rho}$  and *average recall*  $\bar{\varrho}$  are then defined as:

$$\bar{\rho} = \rho \times \alpha, \quad \bar{\varrho} = \varrho \times \beta \quad (5)$$

where  $\rho$  and  $\varrho$  are the precision and recall for our detection functions of the relevant salient objects,  $\alpha$  and  $\beta$  are the classification precision and recall for the classifiers.

**Table 1.** The average performance of some detection functions.

salient objects	brown horse	grass purple	flower red	flower	rock sand	field
$\rho$	95.7%	92.9%	96.1%	87.8%	98.7%	98.8%
$\varrho$	100%	94.8%	95.2%	85.4%	100%	96.6%
salient objects	water human skin	sky	snow sunset/sunrise	waterfall		
$\rho$	86.7%	86.2%	87.6%	86.7%	92.5%	88.5%
$\varrho$	89.5%	85.4%	94.5%	87.7%	95.2%	87.1%
salient objects	yellow flower	forest	sail cloth	elephant	cat	zebra
$\rho$	87.4%	85.4%	96.3%	85.3%	90.5%	87.2%
$\varrho$	89.3%	84.8%	84.9%	88.7%	87.5%	85.4%

The average performance for semantic image classification technique is given in Table 2. They are obtained by averaging *classification accuracy* and *misclassification ratio* over 125,000 Corel images and 35,000 photographs. In order to obtain the real impact of salient objects on semantic-sensitive image content characterization, we compared the performance differences for two image content characterization frameworks by using image blobs and salient objects. One can find that our image content characterization framework by using the salient objects have improved the semantic image classifier accuracy significantly.

**Table 2.** The classification performance (i.e., average precision versus average recall) comparison for our classifiers.

	concept	mountain	view	beach	garden	sailing	skiing	desert
salient	$\bar{\rho}$	81.7%	80.5%	80.6%	87.6%	85.4%	89.6%	
objects	$\bar{\varrho}$	84.3%	84.7%	90.6%	85.5%	83.7%	82.8%	
image	$\bar{\rho}$	73.5%	73.6%	71.3%	72.5%	76.3%	73.6%	
blobs	$\bar{\varrho}$	65.5%	75.9%	68.2%	67.3%	71.2%	68.5%	

Some results of our semantic image classification and multi-level annotation system are given in Fig. 7, where the keywords for automatic image annotation include the multi-level keywords for interpreting both the visually distinguishable salient objects and the relevant semantic image concepts.

6 Conclusions

This paper has proposed a novel framework to achieve a middle-level understanding about image contents by using the salient objects. Based on a novel semantic-sensitive image content representation and semantic image concept interpretation framework, our semantic image classification system has achieved very good performance.



## References

1. A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain, "Content-based image retrieval at the end of the early years", *IEEE Trans. on PAMI*, vol.22, pp.1349-1380, 2000.
2. S.-F. Chang, W. Chen, H. Sundaram, "Semantic visual templates: linking visual features to semantics", Proc. ICIP, 1998.
3. A. Mojsilovic, J. Kovacevic, J. Hu, R.J. Safranek, and S.K. Ganapathy, "Matching and retrieval based on the vocabulary and grammar of color patterns", *IEEE Trans. on Image Processing*, vol.9, pp.38-54, 2000.
4. D.A. Forsyth and M. Fleck, "Body plan", Proc. of CVPR, pp.678-683, 1997.
5. C. Carson, S. Belongie, H. Greenspan, J. Malik, "Blobworld: Image segmentation using expectation-maximization and its application to image querying", *IEEE Trans. PAMI*, vol.24, no.8, 2002.
6. J.Z. Wang, J. Li and G. Wiederhold, "SIMPLIcity: Semantic-sensitive integrated matching for picture libraries", *IEEE Trans. on PAMI*, 2001.
7. W. Wang, Y. Song, A. Zhang, "Semantic-based image retrieval by region saliency", Proc. CIVR, 2002.
8. J.R. Smith and C.S. Li, "Image classification and querying using composite region templates", *Computer Vision and Image Understanding*, vol.75, 1999.
9. P. Lipson, E. Grimson, P. Sinha, "Configuration based scene and image indexing", Proc. CVPR, 1997.
10. A.B. Torralba, A. Oliva, "Semantic organization of scenes using discriminant structural templates", Proc. ICCV, 1999.
11. A. Vailaya, M. Figueiredo, A.K. Jain, H.J. Zhang, "Image classification for content-based indexing", *IEEE Trans. on Image Processing*, vol.10, pp.117-130, 2001.
12. E. Chang, K. Goh, G. Sychay, G. Wu, "CBSA: Content-based annotation for multimodal image retrieval using Bayes point machines", *IEEE Trans. CSVT*, 2002.
13. M. Weber, M. Welling, P. Perona, "Towards automatic discovery of object categories", Proc. CVPR, 2000.
14. J. Luo and S. Etz, "A physical model-based approach to detecting sky in photographic images", *IEEE Trans. on Image Processing*, vol.11, 2002.
15. S. Li, X. Lv, H.J. Zhang, "View-based clustering of object appearances based on independent subspace analysis", Proc. IEEE ICCV, pp.295-300, 2001.
16. A.B. Benitez and S.-F. Chang, "Semantic knowledge construction from annotated image collections", Proc. ICME, 2002.
17. A. Aslandogan, C. Their, C. Yu, J. Zon, N. Rishe, "Image retrieval using WordNet", ACM SIGIR, 1997.
18. X. Zhu and T.S. Huang, "Unifying keywords and visual contents in image retrieval", *IEEE Multimedia*, pp.23-33, 2002.
19. D. Blei, M.I. Jordan, "Modeling annotated data", ACM SIGIR, pp.127-134, 2003.
20. K. Branard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, M.I. Jordan, "Matching words and pictures", *Journal of Machine Learning Research*, vol.3, pp.1107-1135, 2003.
21. Y. Wu, Q. Tian, T.S. Huang, "Discriminant-EM algorithm with application to image retrieval", Proc. CVPR, pp.222-227, 2000.
22. J. Fan, H. Luo, A.K. Elmagarmid, "Concept-oriented indexing of video database: towards more effective retrieval and browsing", *IEEE Trans. on Image Processing*, vol.13, no.5, 2004.

# Multimodal Salient Objects: General Building Blocks of Semantic Video Concepts

Hangzai Luo<sup>1</sup>, Jianping Fan<sup>1</sup>, Yuli Gao<sup>1</sup>, and Guangyou Xu<sup>2</sup>

<sup>1</sup> Department of Computer Science, University of North Carolina  
Charlotte, NC 28223, USA

<sup>2</sup> Department of Computer Science, Tsinghua University  
Beijing, CHINA

**Abstract.** In this paper, we propose a novel video content representation framework to achieve a middle-level understanding of video contents by using *multimodal salient objects*. Specifically, this framework includes: (a) A semantic-sensitive video content representation and semantic video concept modeling framework by using the multimodal salient objects and Gaussian mixture models; (b) A machine learning technique to train the automatic detection functions of multimodal salient objects; (c) A novel framework to enable more effective classifier training by integrating model selection and parameter estimation seamlessly in a single algorithm. Our experiments on a certain domain of medical education videos have obtained very convincing results.

## 1 Introduction

Semantic video classification is a promising approach to achieve more effective video retrieval at the semantic level [1–2, 10–13]. However, the performance of the semantic video classifiers largely depends on two issues: (1) the effectiveness of video patterns for video content representation and feature extraction [1–10]; (2) the significance of the algorithms for semantic video concept modeling and classifier training [11–15].

To address the first issue, most existing CBVR systems select the video shots and homogeneous moving regions as the underlying video patterns for content representation and feature extraction. In addition, most existing CBVR systems only use the shot-based or region-based low-level visual features [4–10]. However, original video is a synergy of multimodal inputs such as audio, vision, and image-text [2]. Thus it is very important to address what kind of *multimodal content integration model* can be used to explore the joint perceptual effects among the multimodal inputs for semantic video concept interpretation [12–15].

Based on these observations, we develop a novel framework to address these problems in a certain domain of **medical education videos**. This paper is organized as follows: Section 2 introduces a novel framework for video content representation and feature extraction; Section 3 introduce a learning-based technique to enable automatic detection of the multimodal salient objects; Section 4 presents a statistical approach to interpret the semantic video concepts via

Gaussian mixture models; Section 5 introduces our experimental results; We conclude in Section 6.

2 Concept-Sensitive Video Content Representation

To achieve a middle-level understanding of video contents, we have developed a novel concept-sensitive video content representation framework by using the *multimodal salient objects*. As shown in Fig. 1, the semantic video concepts are implicitly or explicitly related to some specific multimodal salient objects in a certain video domain. Thus detecting such multimodal salient objects plays an important role on achieving a middle-level understanding of video contents and bridging the semantic gap more effectively.

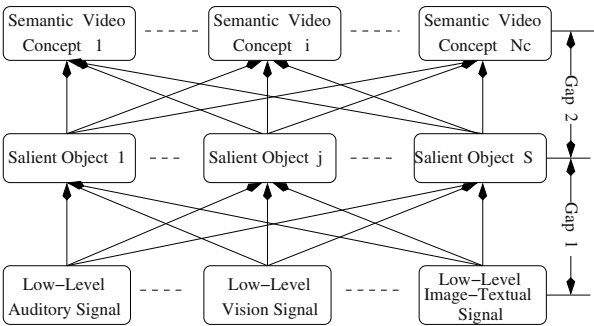


Fig. 1. The proposed video content representation framework.

The multimodal salient objects are defined as the perceptually distinguishable video compounds that are related to human semantics. The basic vocabulary of such multimodal salient objects can be obtained by using the taxonomy of video compounds of medical videos as shown in Fig. 2. In addition, the multimodal salient objects can be categorized into three classes: visual salient objects, auditory salient objects and image-textual salient objects.

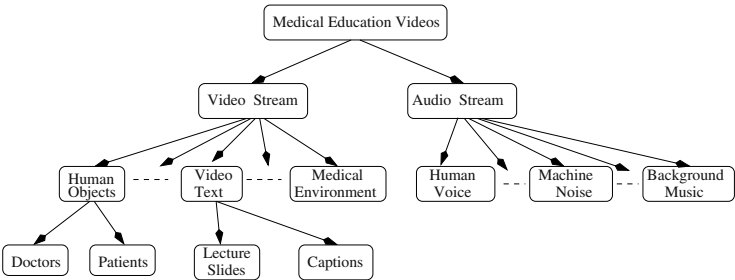
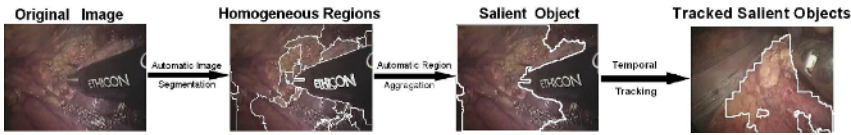


Fig. 2. Examples for compound taxonomy of medical education videos.

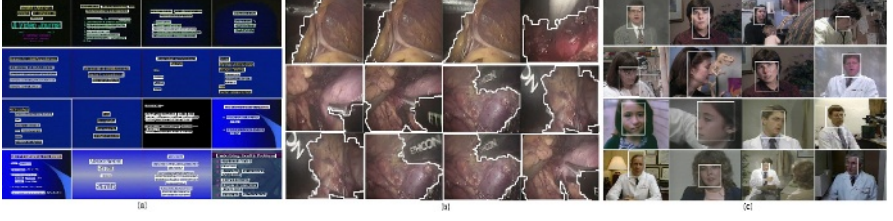
### 3 Detection of Multimodal Salient Objects

To design the automatic detection functions of multimodal salient objects, the video shots are first determined automatically by using adaptive shot detection techniques [4]. The visual features have also been integrated with the auditory features to detect the perceptual content changes among frames [12]. Based on the basic vocabulary of multimodal salient objects as shown in Fig. 2, a set of detection functions have been learned and each function is able to detect one certain type of these domain-dependent multimodal salient objects in the basic vocabulary.



**Fig. 3.** The flowchart for automatic salient object extraction.

We use our detection function for the visual salient object “gastrointestinal regions” as an example to show how we can design our automatic detection functions of multimodal salient objects. Our detection function for the visual salient object “gastrointestinal regions” consists of the following three components as shown in Fig. 3: (a) Image regions with homogeneous color or texture are obtained by using mean shift technique [12]. Since the visual characteristics of a certain type of visual salient objects will look differently at different lighting and capturing conditions [1], using only one video frame is very difficult to represent its characteristics and thus this automatic image segmentation procedure is performed on a set of training video frames with different lighting conditions. (b) The homogeneous image regions, that are explicitly related to the visual salient object “gastrointestinal regions”, are annotated by human interaction. Region-based low-level visual features, such as dominant colors and variances, Tamura textures, object density (i.e., coverage ratio between object region and relevant rectangular box for object representation), height-width ratio for the object rectangular box, are extracted for characterizing the visual properties of these labeled image regions. To generate the detection function for the visual salient object “gastrointestinal regions”, a binary SVM classifier is learned from the labeled samples to classify the image regions into two classes. The connected homogeneous image regions with the same class label are aggregated as the visual salient object “gastrointestinal regions”. (c) The temporal tracking technique is used to integrate the detection results of the visual salient object “gastrointestinal regions” within the same video shot as a single output. Some of our detection results are shown in Fig. 4.



**Fig. 4.** Detection results of multimodal salient objects: (a) lecture slides; (b) gastrointestinal regions; (c) human faces.

## 4 Interpretation of Semantic Video Concepts

To interpret the semantic video concepts, we have also proposed a novel multimodal video context integration framework by using Gaussian mixture models. The class distribution of the multimodal salient objects related to the semantic video concept  $C_j$  is approximated by:

$$P(X, C_j, \Theta_{c_j}) = \sum_{i=1}^{\kappa_j} \omega_{s_i} P(X, C_j | S_i, \theta_{s_i}) \quad (1)$$

where  $\kappa_j$  indicates the optimal number of mixture components,  $\Theta_{c_j} = \{\kappa_j, \omega_{c_j}, \theta_{s_i}, i = 1, \dots, \kappa_j\}$  is the set of the parameters for these mixture components,  $\omega_{c_j} = \{\omega_{s_i}, i = 1, \dots, \kappa_j\}$  is the set of the relative weights among these mixture components,  $\omega_{s_i}$  is the relative weight for the  $i$ th mixture component,  $X = (x_1, \dots, x_n)$  is the  $n$ -dimensional multimodal perceptual features which are used to represent the relevant salient objects.

The central goal of this paper is to automatically determine the optimal number of mixture components and their parameters based on the labeled training samples. We use *one-against-all* rule to label the training samples  $\Omega_{c_j} = \{X_l, C_j(X_l) | l = 1, \dots, N\}$ : positive samples for a specific semantic video concept  $C_j$  and negative samples. Each labeled training sample is a pair  $(X_l, C_j(X_l))$  that consists of a set of multimodal features  $X_l$  and the semantic label  $C_j(X_l)$  for the corresponding multimodal salient objects.

To estimate the optimal number  $\kappa$  of mixture components and their parameters automatically, we have developed an **adaptive EM algorithm** to integrate *parameter estimation* and *model selection* seamlessly in a single algorithm and it takes the following steps:

**Step 1:** Since the maximum likelihood estimate increases as more mixture components are used, a penalty term is added to determine the underlying optimal model structure. Thus the optimal number of mixture components for a certain semantic video concept is estimated by:

$$\hat{\Theta}_{c_j} = \arg \max L(X, \Theta_{c_j}) \quad (2)$$

where  $L(X, \Theta_{c_j}) = \log P(X, C_j, \Theta_{c_j}) + \log p(\Theta_{c_j})$  is the objective function,  $\log P(X, C_j, \Theta_{c_j})$  is the likelihood function as described in Eq. (1), and  $\log p(\Theta_{c_j})$

$= -\frac{n+\kappa_j+3}{2} \sum_{l=1}^{\kappa_j} \log \frac{N\omega_l}{12} - \frac{\kappa_j}{2} \log \frac{N}{12} - \frac{\kappa_j(N+1)}{2}$  is the minimum description length (MDL) term for complexity penalization [12],  $N$  is the total number of training samples and  $n$  is the dimensions of features.

**Step 2:** To avoid the initialization problem, our adaptive EM algorithm starts from a reasonably large value of  $\kappa_j$  to explain all the structure of the data and reduce the number of mixture components sequentially. To escape the local extrema, our adaptive EM algorithm modifies the distribution of the mixture components according to the underlying sample distribution through an automatic *merging* and *splitting* procedure.

We use *Kullback divergence*  $KL(P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l}), P(X, C_j|S_k, \mu_{s_k}, \sigma_{s_k}))$  to measure the divergence between the  $l$ th mixture component  $P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l})$  and the  $m$ th mixture component  $P(X, C_j|S_k, \mu_{s_k}, \sigma_{s_k})$ :

$$KL(P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l}), P(X, C_j|S_k, \mu_{s_k}, \sigma_{s_k})) = \int P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l}) \log \frac{P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l})}{P(X, C_j|S_k, \mu_{s_k}, \sigma_{s_k})} dX \quad (3)$$

If  $KL(P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l}), P(X, C_j|S_k, \mu_{s_k}, \sigma_{s_k}))$  is small, these two mixture components provide strongly overlapped densities and overpopulate the relevant sample regions, thus they can be merged as one new mixture component. In addition, the *local Kullback divergence*  $KL(P(X, C_j|S_{lk}, \mu_{s_{lk}}, \sigma_{s_{lk}}), P(X, C_j|\Theta))$  is used to measure the divergence between the merged mixture component  $P(X, C_j|S_{lk}, \mu_{s_{lk}}, \sigma_{s_{lk}})$  and the local sample density  $P(X, C_j|\Theta)$ . To detect the best candidates for *merging*, our adaptive EM algorithm calculates the local Kullback divergences for  $\frac{\kappa_j(\kappa_j-1)}{2}$  pairs of the mixture components that could be merged. The pair with the minimum value of the local Kullback divergence is selected as the best candidate for potential merging.

At the same time, our adaptive EM algorithm also calculates the *local Kullback divergence*  $KL(P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l}), P(X, C_j|\Theta))$  to measure the divergence between the  $l$ th mixture component  $P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l})$  and the local sample density  $P(X, C_j|\Theta)$ . If the local Kullback divergence for a certain mixture component  $P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l})$  is big, the relevant sample region is underpopulated and the elongated mixture component  $P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l})$  is selected as the best candidate to be *split* into two more representative mixture components.

**Step 3:** If  $KL(P(X, C_j|S_{lk}, \mu_{s_{lk}}, \sigma_{s_{lk}}), P(X, C_j|\Theta)) \leq KL(P(X, C_j|S_l, \mu_{s_l}, \sigma_{s_l}), P(X, C_j|\Theta))$ , merging is performed. Otherwise, splitting is performed. Thus the mixture components are split once their densities grow wide or unusually elongated, while the mixture components are merged once their densities overlap strongly.

**Step 4:** Given the finite mixture model with a certain number of mixture components (i.e., after merging or splitting operation), the EM iteration is performed to estimate their mixture parameters such as means and covariances and weights among different mixture components.

After the EM iteration procedure converges, a weak classifier is built. The performance of this weak classifier is obtained by testing a small number of

labeled samples that are not used for classifier training. If the average performance of this weak classifier is good enough,  $P(C_j|X, \Theta_{c_j}) \geq \delta_1$ , go to step 5. Otherwise, go back step 2.  $\delta_1$  is set to 80% in our current experiments.

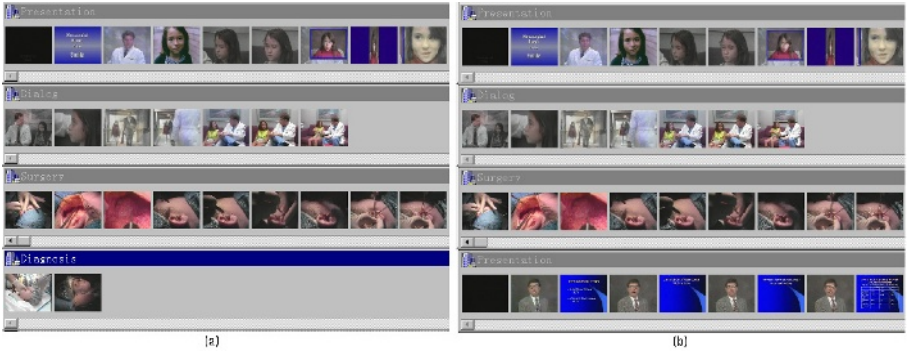
**Step 5:** Output mixture model and parameters  $\Theta_{c_j}$ .

By using an automatic merging and splitting procedure, our adaptive EM algorithm has the following advantages: (a) It does not require a careful initialization by starting with a reasonably large number of the mixture components; (b) It is able to take the advantage of the negative samples for discriminative classifier training; (c) It is able to escape the local extrema and enable a global solution by modifying the distribution of the mixture components in the feature space.

To classify a test video clip  $V_i$ , the salient objects are first detected by using our detection functions. It is important to note that one certain test video clip may consist of multiple salient objects. Thus the test video clip  $V_i$  can be described as a set of the relevant salient objects,  $V_i = \{S_1, \dots, S_l, \dots, S_n\}$ . The class distribution of these salient objects  $V_i = \{S_1, \dots, S_l, \dots, S_n\}$  is then modeled as a Gaussian mixture model  $P(X, C_j, \Theta_{c_j})$ . Finally, the test video clip  $V_i$  with the feature set  $X_i$  is assigned to the best matching semantic video concept  $C_j$  that corresponds to the maximum posterior probability:

$$\text{Max} \left\{ P(C_j|V_i, X_i, \Theta) = \frac{P(C_j)P(X_i, C_j, \Theta_{c_j})}{\sum_{j=1}^{N_c} P(C_j)P(X_i, C_j, \Theta_{c_j})} \right\} \quad (4)$$

where  $\Theta = \{\Theta_{c_j}, j = 1, \dots, N_c\}$  is the set of the mixture Gaussian parameters for the classifier,  $P(C_j)$  is the prior probability of the semantic video concept  $C_j$ . Our current experiments focus on generating five basic semantic medical concepts, “lecture presentation”, “gastrointestinal surgery”, “traumatic surgery”, “dialog”, and “diagnosis”, which are widely distributed in medical education videos. Some semantic video classification results are given in Fig. 5.



**Fig. 5.** The classification results for several test videos that include the semantic video concepts such as “lecture presentation”, “dialog”, “diagnosis”, “traumatic surgery”, “gastrointestinal surgery”.

## 5 Performance Evaluation

Precision  $\rho$  and recall  $\varrho$  are used to measure the average performance of our detection functions:

$$\rho = \frac{\eta}{\eta + \varepsilon}, \quad \varrho = \frac{\eta}{\eta + \vartheta} \quad (5)$$

where  $\eta$  is the set of true positive samples that are related to the corresponding type of salient object and detected correctly,  $\varepsilon$  is the set of true negative samples that are irrelevant to the corresponding type of salient object and detected incorrectly, and  $\vartheta$  is the set of false positive samples that are related to the corresponding type of salient object but mis-detected. The average performances for some detection functions are given in Table 1.

**Table 1.** The average performances (i.e., precision  $\rho$  versus recall  $\varrho$ ) of our detection functions.

salient objects	face	slide	blood	music	speech	skin sketch	dialog	noise	
$\rho$	90.3%	92.4%	90.2%	94.6%	89.8%	96.7%	93.3%	89.7%	86.9%
$\varrho$	87.5%	89.6%	86.7%	81.2%	82.6%	95.4%	88.5%	83.2%	84.7%
salient objects	silence	legs	stomach	captions	blue cloth	colon			
$\rho$	96.3%	92.4%	89.7%	91.8%	96.7%	94.3%			
$\varrho$	94.5%	89.8%	91.2%	93.2%	95.8%	87.5%			

The *benchmark metric* for classifier evaluation includes *classification precision*  $\xi$  and *classification recall*  $\zeta$ . The classification precision  $\xi$  and recall  $\zeta$  are defined as:

$$\xi = \frac{\varphi}{\varphi + \psi}, \quad \zeta = \frac{\varphi}{\varphi + \varsigma} \quad (6)$$

where  $\varphi$  is the set of true positive samples that are related to the corresponding semantic video concept and classified correctly,  $\psi$  is the set of true negative samples that are irrelevant to the corresponding semantic video concept and classified incorrectly,  $\varsigma$  is the set of false positive samples that are related to the corresponding semantic video concept but mis-classified.

As mentioned above, two key issues may affect the performance of the semantic video classifiers: (a) the performance of our detection functions of salient objects and the quality of features on discriminating among different semantic video concepts; (b) the performance of the algorithms for classifier training. Thus the real impacts of semantic video classifiers come from these two key issues, and the benchmark metric for performance evaluation is then defined as:

$$\bar{\rho} = \rho \times \xi, \quad \bar{\varrho} = \varrho \times \zeta \quad (7)$$

where  $\rho$  and  $\varrho$  are the precision and recall for our detection functions of the relevant salient objects,  $\xi$  and  $\zeta$  are the classification precision and recall for the classifiers.

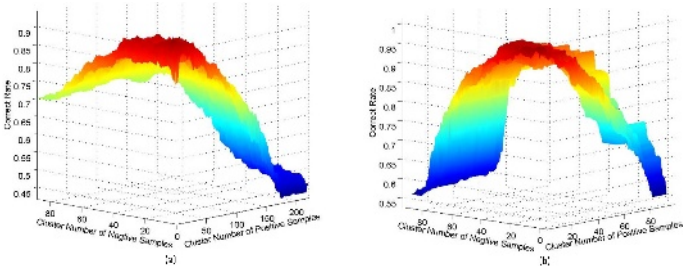


To evaluate the performance of our semantic video classifiers, we have compared their performance differences by using different frameworks for video content representation and feature extraction: salient objects *versus* video shots. The average performance differences are given in Tables 2, they are obtained by averaging *precision* and *recall* for the same semantic video concept over 35,400 test video clips. To show the real impact of using salient objects for video content representation, we use  $\bar{\rho} = \rho \times \xi$  and  $\bar{\varrho} = \varrho \times \zeta$  as the benchmark metric. From our results, we have found that using salient objects for video content representation and feature extraction can improve the classifier performance significantly. The reason is that the salient objects are concept-sensitive and thus the multimodal features extracted from the salient objects are more effective to discriminate among different semantic video concepts.

**Table 2.** The average performance differences (i.e.,  $\bar{\rho} = \rho \times \xi$  versus  $\bar{\varrho} = \varrho \times \zeta$ ) for the SVM classifiers under different video content representation schemes.

concepts	lecture	trauma	diagnosis	gastrointestinal	dialog	burn
	surgery		surgery		surgery	
salient $\bar{\rho}$	81.6%	82.3%	81.7%	85.1%	79.2%	81.4%
objects $\bar{\varrho}$	82.1%	81.9%	89.3%	89.3%	78.6%	85.2%
video $\bar{\rho}$	64.9%	64.3%	61.1%	71.5%	67.5%	64.8%
shots $\bar{\varrho}$	68.5%	65.7%	59.7%	69.3%	68.2%	69.1%

Experimentally, we have also studied the convergence of our adaptive EM algorithm and the result is given in Fig. 6. One can find that the classifier’s performance increases step by step before our adaptive EM algorithm converges to the underlying optimal model of a certain semantic video concept. After our adaptive EM algorithm converges to the underlying optimal model, adding more mixture components to the Gaussian mixture model decreases the classifier’s performance significantly.



**Fig. 6.** The correct ratio of classification versus the underlying model structure  $\kappa$  for medical education video classification: (a) lecture presentation; (b) traumatic surgery.

Our current experiments are done by using 2G *Hz* EDLL PC with Pentium4. We have trained 6 classifiers for six semantic video concepts simultaneously and we set the maximum value of  $\kappa = 100$  for searching their optimal models. For each semantic video concepts, we have labeled 150 positive samples and totally we have 900 labeled samples for 6 semantic video concepts. The positive samples for a certain semantic video concept can be taken as the negative samples for other semantic video concepts.

## 6 Conclusions and Future Works

In this paper, we propose a novel framework to achieve more effective semantic-sensitive video content characterization and indexing by using *multimodal salient objects*. In addition, a novel framework to interpret semantic video concepts by using Gaussian mixture model. Our experiments on a certain domain of medical education videos have obtained very convincing results.

It is worth noting that the proposed classifier training techniques can also be used for other video domains when the labeled training samples are available. Our future works will focus on extending our semantic video classification techniques to other video domains such as news and films.

## References

1. S.-F. Chang, W. Chen, H. Sundaram, "Semantic visual templates: linking visual features to semantics", Proc. ICIP, 1998.
2. W.H. Adams, G. Iyengar, C.-Y. Lin, M.R. Naphade, C. Neti, H.J. Nock, J.R. Smith, "Semantic indexing of multimedia content using visual, audio and text cues", *EURASIP JASP*, vol.2, pp.170-185, 2003.
3. D.A. Forsyth and M. Fleck, "Body plan", Proc. of CVPR, pp.678-683, 1997.
4. H.J. Zhang, J. Wu, D. Zhong and S. Smoliar, "An integrated system for content-based video retrieval and browsing", *Pattern Recognition*, vol.30, pp.643-658, 1997.
5. S. Satoh and T. Kanada, "Name-It: Association of face and name in video", Proc. of CVPR, 1997.
6. S.F. Chang, W. Chen, H.J. Meng, H. Sundaram and D. Zhong, "A fully automatic content-based video search engine supporting spatiotemporal queries", *IEEE Trans. on CSVT*, vol.8, pp.602-615, 1998.
7. Y. Deng and B.S. Manjunath, "Netra-V: Toward an object-based video representation", *IEEE Trans. on CSVT*, vol.8, pp.616-627, 1998.
8. N. Dimitrova, H.J. Zhang, B. Shahraray, I. Sezan, T.S. Huang, A. Zakhori, "Applications of video-content analysis and retrieval", *IEEE Multimedia*, pp.42-55, 2002.
9. Y. Rui, T.S. Huang and S.F. Chang, "Image retrieval: Past, present, and future", *Journal of Visual Comm. and Image Represent.*, vol.10, pp.39-62, 1999.
10. M. Lew, *Principals of Visual Information Retrieval*, Springer-Verlag, 2001.
11. K. Branard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, M.I. Jordan, "Matching words and pictures", *Journal of Machine Learning Research*, vol.3, pp.1107-1135, 2003.

12. J. Fan, H. Luo, A.K. Elmagarmid, "Concept-oriented indexing of video database: towards more effective retrieval and browsing", *IEEE Trans. on Image Processing*, vol.13, no.5, 2004.
13. A.B. Benitez, J.R. Smith and S.-F. Chang, "MediaNet: A multimedia information network for knowledge representation", *Proc. SPIE*, vol.4210, 2000.
14. M.R. Naphade and T.S. Huang, "A probabilistic framework for semantic video indexing, filtering, and retrieval", *IEEE Trans. on Multimedia*, vol.3, pp.141-151, 2001.
15. S. Paek, C. Sable, et al., "Integration of visual and text-based approaches for the content labeling and classification of photographs", *SIGIR Workshop on MIR*, 1999.
16. Y. Wu, Q. Tian, T.S. Huang, "Discriminant-EM algorithm with application to image retrieval", *Proc. CVPR*, pp.222-227, 2000.

# Video Content Representation as Salient Regions of Activity

Nicolas Moënné-Loccoz, Eric Bruno, and Stéphane Marchand-Maillet

University of Geneva 24, rue General Dufour - 1211 Geneva 4, Switzerland

{Nicolas.Moenne-Loccoz, Eric.Bruno,  
Stephane.Marchand-Maillet}@cui.unige.ch,  
<http://viper.unige.ch/>

**Abstract.** In this paper, we present a generic and robust representation of video shots content expressed in terms of salient regions of activity. The proposed approach is based on salient points of the image space, thus minimizing the computational effort. Salient points are extracted from each frame. Their trajectories are computed between successive frames and the global motion model is estimated. Moving salient points are selected from which salient regions are estimated using an adaptive Mean-Shift process, based on the statistical properties of the point neighborhoods. The salient regions are then matched along the stream, using the salient points trajectories. The information carried by the proposed salient regions of activity is evaluated and we show that such a representation of the content forms suitable input for video content interpretation algorithms.

## 1 Introduction

Video content representation is the task of extracting information from a video stream in order to facilitate its interpretation. We base our study on video shots as they are temporal segments of the video with consistent visual content. The information carried by video shots is twofold. First, a video shot holds a visual context, i.e. the visual environment and the editing effects. A video shot also contains events, i.e. long-term temporal objects [16]. We focus our approach on the extraction of such events and aim at a well defined representation of the visual content.

To extract events occurring in video shots, most approaches either rely on domain-specific algorithms or on global features. Works using local features in an unconstrained capturing environment are mainly based on spatio-temporal segmentation of the entire scene, which is a difficult problem that has no generic solution. As the activity of a scene is limited to a given number of regions, we propose to extract salient regions of activity. By definition, these are salient regions of the scene having homogenous perception properties and a salient temporal behavior. This paper presents a robust and fast process to extract salient regions of activity derived from spatio-temporal salient points. A representation

of the video content is obtained as the set of the salient regions defined in the 2D+T space of the video stream. Salient regions of activity highly correlate with the events occurring in the considered scene. As such, they allow the detection of events that can be used by video content interpretation algorithms.

Section 2 describes the spatio-temporal salient points extraction process. Multi-scale salient points are first extracted from each frame of the video stream. Then, salient points are matched between successive frames in order to estimate their trajectory. The global affine motion model is estimated from the trajectories of the points. Spatio-temporal salient points are selected as the points that do not follow the global motion. Section 3 details how salient regions are estimated from moving salient points using an Adaptive Mean-Shift process that is based on the color distribution of the point neighborhoods. The extracted salient regions of activity are matched along the stream by considering the trajectories of the points belonging to it. Section 4 provides an evaluation of the information carried by such a representation, demonstrating its robustness and validating its usefulness in the context of video content interpretation. Finally section 5 discusses the main contributions of the present work.

## 2 Spatio-temporal Salient Points

### 2.1 Spatial Salient Points

Salient points are points in the image space where the intensity changes in at least two directions ; because they carry high information about the structural content of a scene, they are used for image retrieval [14,4], imaging parameters estimation [1,17] and objects recognition [8]. Several algorithms have been proposed to extract salient points such as the *Plessey feature points* [5], *SUSAN* [12], *Curvature Scale Space* [10] and the Wavelet based [6] approach.

K. Mikolajczyk and C. Schmid in [9] present a multi-scale version of the Plessey feature points. Their points are among the most robust and have the desirable property to be scale invariant. The multi-scale interest points are extracted from the scale space of an intensity image  $I(v)$ ,  $v \in V = \{x, y\}$ . The scale-space is obtained by convolving  $I(v)$  with a Gaussian derivative kernel for a set of scales  $s \in S$  :

$$L_{v_i}(v, s) = I(v) \star G_{v_i}(s), \forall v \in V, \forall s \in S \quad (1)$$

where  $G_{v_i}$ ,  $i \in \{1, 2\}$  is the Gaussian derivative along the dimension  $v_i$  of the image space  $V$ .

The scale normalized Harris function  $H(v, s)$  is computed at each scale  $s$  and each location  $v$  :

$$H(v, s) = \text{Det}(\Sigma(v, s)) - \alpha \text{Trace}^2(\Sigma(v, s)) \quad (2)$$

$$\Sigma(v, s) = s^2 G(v, \sigma) \star \begin{pmatrix} L_{v_1}^2(v, s) & L_{v_1} L_{v_2}(v, s) \\ L_{v_1} L_{v_2}(v, s) & L_{v_2}^2(v, s) \end{pmatrix} \quad (3)$$

$H(v, s)$  gives a measure of the cornerness of the points  $v$  at the scale  $s$ . It is based on the strength of the eigenvalues of the auto-covariance matrix  $\Sigma$  of  $L_v$ . According to the scale-space theory [7], the characteristic scale of a point is a local maxima of the Laplacian defined by :

$$Lp(v, s) = s^2 |L_{v_1 v_1}(v, s) + L_{v_2 v_2}(v, s)| \quad (4)$$

Thus, multi-scale salient points are defined to be points that are local maxima of  $H$  in the image space and local maxima of  $L$  in the scale space.

Such interest points defined by K. Mikolajczyk and C. Schmid are extracted for each frame of the video stream. In the sequel, we note  $W_t$  the set of salient points  $w$  extracted independently at the frame  $F_t$  and  $s_w$  the characteristic scale of point  $w$ .

## 2.2 Temporal Salient Points

Spatial salient points are located on the objects of interest but also on textured surfaces or non-informative background structures. For these reasons, we consider only the moving salient points, i.e. points which motion is different from that of the background. To extract these temporal salient points, trajectories of the spatial salient points are computed and the background motion model is estimated. Temporal salient points are selected as the spatial salient points having a salient temporal behavior for a given period of time.

**Salient points trajectories.** The trajectories of salient points are computed by matching them between two consecutive frames. More formally, for any  $w_t \in W_t$ , the corresponding point  $w_{t-1} \in W_{t-1}$  is selected. Points in  $W_t$  and  $W_{t-1}$  are first described by a the 9-dimension vector of the Hilbert invariants up to the third order as presented by C. Schmid in [11]. Then, the Mahalanobis distance  $d(w_t, w_{t-1})$  is computed for each pair of points. To select the good matches  $M_t$  among the set of possible matches, a greedy algorithm is applied that tends to minimize the sum of the distances  $\sum_{(w_i, w_j) \in M_t} d(w_i, w_j)$ .

We therefore obtain the set of matches  $M_t$  that associates the salient points of  $W_t$  in the current frame with their predecessors in the previous one. The set of match  $M_t$  corresponds to the set of trajectories of the points between the two successive frames  $F_{t-1}$  and  $F_t$ .

**Global motion model estimation.** Given the set of trajectories  $M_t$ , we estimate the most representative affine motion model of the trajectories (see [13] for an overview of motion estimation). This model thus corresponds to a global description of the background motion. We choose the affine motion model because of its ability to capture the main camera motions with a limited number of parameters:

$$d(v) = \begin{pmatrix} a_1 & a_2 \\ a_4 & a_5 \end{pmatrix} v + \begin{pmatrix} a_3 \\ a_6 \end{pmatrix}$$

To estimate the motion model from the set of trajectories  $M_t$ , we first apply a RanSaC algorithm [3] which tends to select the most representative motion model. As the set of trajectories contains noise, the model is then smoothed by applying a *Tukey M*-estimator in way close to the one presented in [15].

**Points selection.** Moving salient points are those that do not follow the background motion model. In order to remove potential noise, only salient points detected as moving for a given time interval are selected (in our experiments we set this interval to 10 frames). Such points are mainly situated on moving objects and provide high information about the dynamic content of the scene. However, as they are located on corner like regions, they do not well focus on the visual events.

### 3 Salient Regions of Activity

Salient regions of activity are homogenous (in terms of perception properties, e.g. color distribution) moving regions defined in the 2D+T space of the video shot. They tend to correspond to the events (as defined in [16]) occurring in the scene. We detect and track such regions from the moving salient points. Hence, they inherit the saliency and the robustness of such features. We model salient regions of activity as a succession in time of spatial ellipses. The parameters of the ellipses are estimated by a *Mean Shift* algorithm augmented with a kernel adaptation step.

#### 3.1 Adaptive Mean Shift Algorithm

The Mean Shift algorithm has been used to track regions of interest [2]. The main idea is to compute an offset  $\delta_{v_r}$  between a current estimation of the region location  $v_r$  and an estimation  $v'_r$  having a higher likelihood. The offset is computed by :

$$\delta_r = \frac{\sum_v K(v - v_r) p(v) (v - v_r)}{\sum_v K(v - v_r) p(v)}$$

where  $K$  is a kernel centered in  $v_r$  and  $p(v)$  is a weighting function measuring the probability of the pixel  $v$  to belong to the region. From an initial location, the algorithm computes a new location from the offset  $\delta_{v_r}$  and is iterated until the location converges to a local maxima.

In our setting, we use an ellipsoidal Epanechnikov Kernel because it has a convergence rate much more higher than the Gaussian Kernel :

$$K(v - r) = \begin{cases} \frac{3}{4} \left( 1 - \left( \frac{(v - v_r)^T \Sigma_k (v - v_r)}{\sigma_s} \right)^2 \right) & \text{if } \left| \frac{(v - v_r)^T \Sigma_k (v - v_r)}{\sigma_s} \right| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

where  $\Sigma_k$  is the affine transformation corresponding to the shape of the kernel and  $\sigma_s$  is the size of the kernel (its radius in the affine transformed domain). As the shape and the size of the salient regions of activity are not known a priori, a shape adaptation step is necessary after the convergence of the Mean Shift. The shape and the size of the kernel are estimated from the covariance matrix of  $p(v)$ .

The algorithm then alternates a Mean Shift algorithm with an adaptation step until both the location of the region  $v_r$  and the kernel converge. In order to speed up the process, some divergence criteria may be defined such as a maximum size of the Kernel and a minimum sum of the weights  $p(v)$ .

### 3.2 Region Detection

A new salient region of activity is detected for each moving salient point not within a currently tracked region. The initial position of the region is set to the position of the corresponding salient point :  $v_r = w$ . The shape of the kernel is initialized to a circle, i.e.  $\Sigma_k = \text{diag}(1, 1)$  and its size to its characteristic scale, i.e.  $\sigma_s = 3 * w_s$ .

As weighting function, we use  $P(v|\theta_w) = N(\mu_{\theta_w}, \Sigma_{\theta_w})$ , the probability of a pixel to be in the Gaussian *RGB* normalized ( $R/(R + G + B), G/(R + G + B)$ ) color space defined by  $\theta_w$ . The parameters  $\theta_w$  are estimated in the spatial neighborhood of the salient point (i.e., the neighborhood inside the circle of radius  $3 * w_s$  centered at the point location).

An adaptive mean shift process is then performed that estimates the ellipse maximizing the likelihood of the region.

### 3.3 Region Tracking

Detected salient regions of activity are tracked along the shot. First, the position of a region is updated by adding to its previous position the mean motion vector of the salient points within it :  $v_r^{t+1} = v_r^t + E[(w_{t+1} - w_t)]$ . Then, an adaptive mean shift process is performed which updates the ellipse estimate.

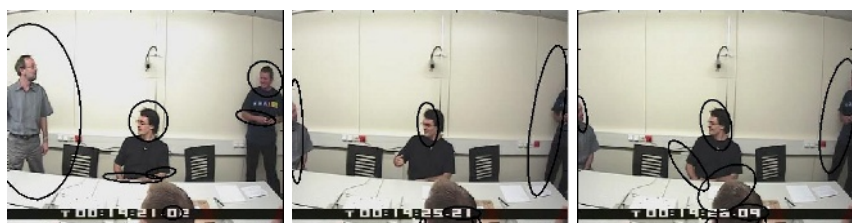
Figure 1 presents examples of detected and tracked salient regions in different settings.

## 4 Evaluation

In order to evaluate the information provided by the extracted salient regions of activity we consider the events they characterize and evaluate the *precision* of this information. This study is performed over a corpus containing about 50 shots: 22 *sport* shots, 13 *news* shots, 14 *sitcom* shots (from the *MPEG7 evaluation corpus*) and 5 videos of meeting (from the *M4 meeting corpus*).

We have labeled about 1000 extracted salient regions according to the events they characterize (see table 1). We observe a precision above **63%**. In other





Meeting room (static camera)



News (static camera)



News (moving camera)



Basketball game (moving camera)



Soccer game (moving camera)

**Fig. 1.** Samples of Salient Regions of Activity.

words, almost 2 out of 3 regions are semantically meaningful. Hence, salient regions of activity are well-suited to provide video interpretation algorithms with a meaningful decomposition of the spatio-temporal content in terms of events.

**Table 1.** Salient regions labels.

Number of regions	Label
312	Moving Human Body
122	Moving Head
56	Moving Arm
62	Moving Leg
24	Agitated Group of Person
45	Text Overlay
9	Moving Ball
364	Noise
994	Total

Table 2 shows a clear correlation between the labels of the detected salient regions of activity and the class of the video shot. For example, salient regions extracted in *sport action* sequences, correspond mainly to *Moving Human body* and *Moving Leg* (running player). For *Speaking Anchor Person* sequences, regions mostly correspond to *Moving Head* and *Text Overlay* (anchor person and textual information). Thus, these results show that salient regions of activity are consistent with the content of the shot and form cues that could be used to perform event retrieval or recognition.

**Table 2.** Labels repartition according to the shot class.

	Sport Action	Speaking Anchor Person	Meeting Action
Moving Human Body	<b>39.09</b>	9.60	<b>25.82</b>
Moving Head	4.2	<b>27.40</b>	<b>30.05</b>
Moving Arm	3.87	12.80	8.45
Moving Leg	<b>10.01</b>	0.00	0.00
Agitated Group of Person	3.87	0.00	0.00
Text Overlay	4.36	<b>16.10</b>	0.00
Moving Ball	1.45	0.00	0.00
Noise	33.11	33.80	35.68
Total	100	100	100

The *recall* of this representation cannot be evaluated in a systematic way because it calls for the exhaustive annotation of all events (defined as long-term temporal objects) of the corpus performed by naive subjects. However,

a systematic visual inspection of the spatio-temporal location of the detected regions let us envisage that they highlight most of the events and that they could be used by higher-level algorithms as a focus of attention preprocessor. Figure 1 shows some visual illustrative examples.

## 5 Conclusion

In this paper, we have presented a novel representation of video content that captures the events occurring in the stream. The proposed procedure for salient regions of activity extraction is fast (less than 1 second per frame in our matlab implementation) robust (based on invariant features) and generic (it makes no assumption on the type of content at hand).

Our results clearly demonstrate that this representation is coherent (i.e. robust and meaningful) and is an effective decomposition of the content that can be further exploited by interpretation processes.

Future work will therefore focus on the use of such a representation to develop new event retrieval algorithms in videos databases.

**Acknowledgments.** This work is funded by EU-IST project M4 ([www.m4project.org](http://www.m4project.org)) and the Swiss NCCR IM2 (Interactive Multimodal Information Management).

## References

1. A. Baumberg. Reliable feature matching across widely separated views. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 774–781, 2000.
2. D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. pages 142–151, 2000.
3. M.A. Fisher and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. In *Communications of the ACM*, pages 381–395, 1981.
4. V. Gouet and N. Boujemaa. On the robustness of color points of interest for image retrieval. In *IEEE International Conference on Image Processing ICIP'2002*, 2002.
5. C. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 189–192, 1988.
6. J.-S. Lee, Y.-N. Sun, and C.-H. Chen. Multiscale corner detection by using wavelet transform. In *IEEE Transactions on Image Processing*, 1995.
7. T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):77–116, 1998.
8. David G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Conference on Computer Vision ICCV, Corfu*, pages 1150–1157, 1999.
9. Krystian Mikolajczyk and Cordelia Schmid. Indexing based on scale invariant interest points. In *8th International Conference on Computer Vision*, pages 525–531, 2001.

10. F. Mokhtarian and R. Suomela. Robust image corner detection through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, 20(12):1376–1381, 1998.
11. C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, 19(5), 1997.
12. S. M. Smith and J. M. Brady. SUSAN – A new approach to low level image processing. Technical Report TR95SMS1c, Chertsey, Surrey, UK, 1995.
13. C. Stiller and J. Konrad. Estimating motion in image sequences: A tutorial on modeling and computation of 2D motion. *IEEE Signal Process*, 16:70–91, 1999.
14. Q. Tian, N. Sebe, M. S. Lew, E. Loupilas, and T. S. Huang. Image retrieval using wavelet-based salient points. In *Journal of Electronic Imaging, Special Issue on Storage and Retrieval of Digital Media*, pages 835–849, 2001.
15. Philip H. S. Torr and Andrew Zisserman. Feature based methods for structure and motion estimation. In *Workshop on Vision Algorithms*, pages 278–294, 1999.
16. L. Zelnik-Manor and M. Irani. Event-based analysis of video. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.
17. Zhengyou Zhang, Rachid Deriche, Olivier D. Faugeras, and Quang-Tuan Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1-2):87–119, 1995.

# Image Classification into Object / Non-object Classes

Sungyoung Kim<sup>1</sup>, Sojung Park<sup>2</sup>, and Minhwan Kim<sup>2</sup>

<sup>1</sup> School of Computer Engineering, Kumoh National Institute of Technology, Gumi, Korea  
sykim@kumoh.ac.kr

<sup>2</sup> Dept. of Computer Engineering, Pusan National Univ., Pusan, Korea  
{sokkobi, mhkim}@pusan.ac.kr

**Abstract.** We propose a method that automatically classifies the images into the object and non-object images. An object image is an image with object(s). An object in an image is defined as a set of regions located near the center of the image, which has significant color distribution compared with its surrounding (or background) region. We define three measures for the classification based on the characteristics of an object. The center significance is calculated from the difference in color distribution between the center area and its surrounding region. Second measure is the variance of significantly correlated colors in the image plane. Significantly correlated colors are first defined as the colors of two adjacent pixels that appear more frequently around center of an image rather than at the background of the image. The last one is the edge strength at the boundary of the region that is estimated as an object. To classify the images we combine each measure by training the neural network. A test with 900 images shows a classification accuracy of 84.2%. We also compare our result with the performance of several other classifiers, Naïve Bayes, Decision Table, and Decision Tree.

## 1 Introduction

In content-based image retrieval (CBIR), images are automatically indexed by summarizing their visual contents, and are searched and matched usually based on low-level features such as color, texture, shape, and spatial layout. However, we know that there is obvious semantic gap between what user-queries represent based on the low-level image features and what the users think. To overcome the semantic gap, many researchers have investigated techniques that retain some degree of human intervention either during input or search thereby utilizing human semantics, knowledge, and recognition ability effectively for semantic retrieval. These techniques called relevance feedbacks are capable of continuous learning through run-time interaction with end-users. Semantic feature finding approaches have been also studied, which tried to extract semantic information directly from images. Automatic classification of scenes into general types such as indoor/outdoor or city/landscape [1-3] is an example of utilizing the semantic information.

On the one hand, many researchers believe that the key to effective CBIR performance lies in the ability to access images at the level of objects because users generally want to search for the images containing particular *object(s) of interest*. Thus, it may be an important step in CBIR to determine whether there is an object(s)

in an image. An object / non-object image classification can improve retrieval performance by filtering out images that are classified as another class. For example, when a user wants to retrieve 'tiger' images, the category to be retrieved can be restricted to the object image class only. The object / non-object image classification can be also utilized for the pre-process of object-based applications, such as the extraction of objects from object class images [4, 5] and the classification of object types to improve image retrieval performance [6].

An object image can be characterized by the presence of object(s). However, determining an *object of interest* in an image is an unresolved issue [5]. It is subjective to determine the object of interest on which user's attention is concentrated. For example, a reddish rising sun with a significant size and salient color distribution in an image can be regarded as an object. However, a meridian sun which has a small size in the image corner may not be an object. The sun may not be an object of interest when a sun is treated as an object. Then, is the sunrise image an object image? The sunrise image is assumed to be an object image when the sun in the image is treated as an object.

In this paper, we propose a method that automatically classifies images into the object and non-object image classes. Generally an object in an image can be defined as a region located near the center of the image, which is characterized as having significant color distribution compared with its surrounding region. The characteristic region tends to be not elongated and to have strong boundary against the surrounding region. Thus we defined three measures based on the characteristics of an object in an image. The first measure is called a center significance that indicates color significance of image center area against its surrounding region. The center area is defined as the region in the center 25% of an image [4]. The second measure is to describe horizontal or vertical distribution of significant pixels in image plane. The significant pixels indicate the pixels that have significant color. Their horizontal and vertical variances are first computed and the larger one is defined as the second measure. The third measure is defined as average edge strength of boundary pixels of a central object [4]. The neural network technique is used for combining the three measures effectively.

## 2 Definition of Object

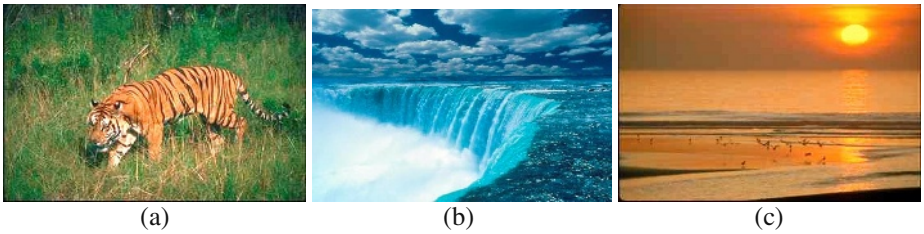
An object in a color image tends to satisfy the following conditions.

- (1) It is located near center of the image,
- (2) It has significant color or texture characteristics against its surrounding area,
- (3) Its size is relatively big,
- (4) Its boundary pixels have relatively strong edginess,
- (5) It has restricted shape, and
- (6) It is related to a specific semantic key word.

It was shown that conditions (1)-(4) were useful to describe objects in ref. [4]. Through the evaluation on the location and size of the manually extracted objects, it was showed that the objects of interest are often located near the center of image. The objects could be displaced from the image center to one of image border directions by some extent. On the one hand, an object with no salient color distribution in a low resolution image or protective colors could be discarded.

There are two additional conditions to clearly distinguish object images from non-object images. Condition (5) discards some objects whose shapes can be temporarily changeable because they are not interesting objects to us usually. For example, we are not interested in clouds in an image. Condition (6) is obvious even though it cannot be measured easily.

Fig. 1(a) and (b) show the objects that can be assigned with key words, tiger and waterfall, respectively. Note that the tiger is clearly a meaningful object. Furthermore, the waterfall can be object because it is a salient region and satisfies our conditions. Thus these two images can be classified into object class image. However, the image in Fig. 1(c) does not have any meaningful region in center of the image, so it is a non-object image. On the one hand, the waterfall in Fig. 1(b) may not be an object of interest, because our attention is not restricted to a specific area but to the whole image, as in Fig. 1(c).



**Fig. 1.** This figure shows examples of object and non-object image. The left and center images can be regarded as object images, while the right image cannot. On the one hand, the left image contains an object of interest but the center image does not

Concept of object of interest is very important in CBIR because users generally want to search for the images containing particular objects of interest. However, it is difficult to describe the objects of interest using only the low-level image features. Thus our research focuses on determining the existence of object rather than the object of interest. We will describe three measures for characterizing object images in next section.

### 3 Classification of Object and Non-object Images

#### 3.1 Significant Features in the Default Attention Window

A significantly correlated color [4] is defined as the color pair  $(c_i, c_j)$  that satisfies Eq. (1), where  $C_{DAW}(c_i, c_j)$  and  $C_{SR}(c_i, c_j)$  are the count of  $(c_i, c_j)$  in the correlogram  $C_{DAW}$  for the default attention window (DAW) and one in  $C_{SR}$  for the surrounding region, respectively. The DAW is defined as the rectangle that is located at the center of an image and whose height and width are set to half the image height and width, respectively. The region outside of the DAW is called the surrounding region in this paper. Significant pixels for a significantly correlated color  $(c_i, c_j)$  are defined as the adjacent pixels in the image one of which has the color  $c_i$  and the other the color  $c_j$ .

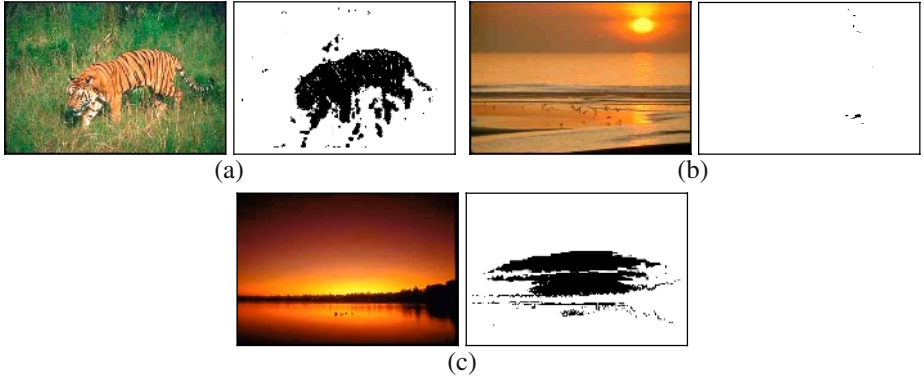
$$\frac{C_{DAW}(c_i, c_j) - C_{SR}(c_i, c_j)}{C_{DAW}(c_i, c_j)} \geq 0.1 \quad (1)$$

### 3.2 Measures for Classification

#### 3.2.1 Center Significance

Center significance of an image is defined as in Eq. 2, which represents significance of color distribution in DAW against the surrounding region. It tends to increase in proportion to density of significant pixels near center of the image. In Fig. 2(a), the object of interest, tiger, presents dense significant pixels and a large center significance value. We can expect to obtain a small center significance value for the non-object image Fig. 2(b). However the center significance does not work well for a non-object image whose background color changes gradually as in Fig. 2(c).

$$CS = \frac{\sum_i \sum_j \text{Max}(C_{DAW}(i, j) - C_{SR}(i, j), 0)}{\sum_i \sum_j C_{DAW}(i, j)} \quad (2)$$



**Fig. 2.** First image shows dense significant pixels near the center with the help of the ‘tiger’ and has a large center significance value (0.466). Second image has a small center significance value (0.002) because it does not contain any objects. Third non-object image has a large center significance value (0.548) because its background color changes gradually near the center

#### 3.2.2 Variance of Significant Pixels

Significant pixels in non-object images tend to be distributed with large variance, while those in object images with small variance. Thus variance of significant pixels in horizontal or vertical direction can be used as a measure of distinguishing object images from non-object images. This measure is defined as the larger variance between the horizontal variance  $V_x$  and the vertical one  $V_y$  in Eq. 3.  $P_x$  and  $P_y$  are the  $x$  and  $y$  coordinates of each significant pixel in an image, respectively.  $W$  and  $H$  represent the horizontal and vertical image size, respectively.  $N$  is the number of significant pixels in the image.



$$V_x(SP) = \frac{\sum \left( (P_x - m_x)^2 / W \right)}{N}, \quad V_y(SP) = \frac{\sum \left( (P_y - m_y)^2 / H \right)}{N} \quad (3)$$

$$V(SP) = \text{Max}(V_x(SP), V_y(SP))$$

### 3.2.3 Edge Strength at Object Boundary

Another measure is the edge strength at the boundary of a central object in an image. The central object is determined by using the extraction method in [4]. The method can extract objects of interest well, but it sometimes extracts meaningless regions when it is applied to non-object images. The edge strength in object images tends to be strong, while one in non-object images relatively weak. If the method [4] cannot extract any central region, then the image is classified as a non-object class image. The edge strength measure is defined as in Eq. (4), where  $\nabla f_i$  represents edge strength of the  $i$ -th boundary pixel and  $N$  is the total number of boundary pixels.

$$E(CO) = \frac{\sum_i \nabla f_i}{N} \quad (4)$$

### 3.3 Neural Network Classifier

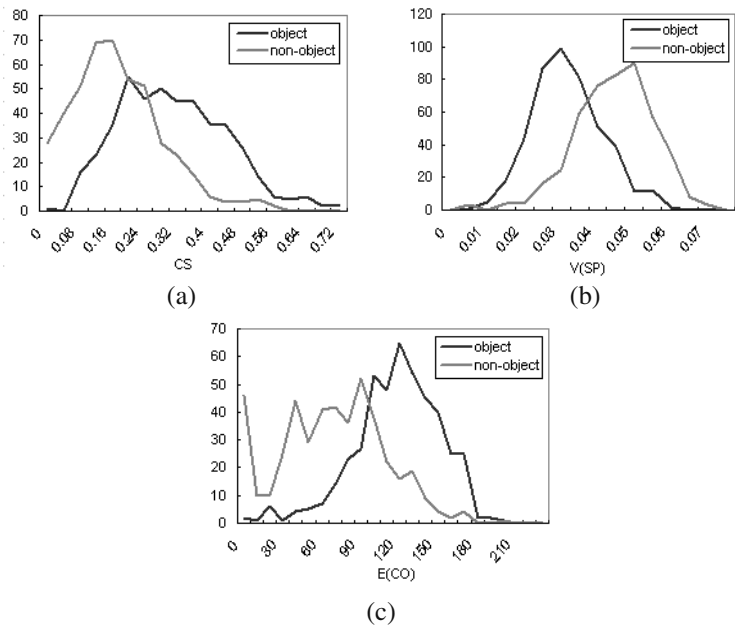
Even though each measure by itself may be useful to classify images, the classification accuracy can be improved by combining three measures. We have used a neural network [8] to optimally combine them, since it was hard to assign an appropriate weight to each measure. We adopt the back-propagation algorithm because of its simplicity and reasonable speed.

## 4 Experimental Results and Discussions

The proposed method is evaluated on 900 images selected from the Corel photo-CD, which consist of 450 object images and 450 non-object images. To verify the distinguishing capability of each measure for classification, we check up the distribution of values for the object and non-object images when each measure is applied to them. Fig. 3(a)-(c) show the distribution curves for the center significance, the variance of significant pixels, and the edge strength measure, respectively. We can see that two curves in each figure are not clearly separated. Thus each measure does not provide very good performance by itself.

The back-propagation neural network is trained using the 6-fold cross-validation [9] to mitigate bias caused by the particular sample chosen for holdout. In  $n$ -fold cross-validation, the data is partitioned into  $n$  equal fold and each fold in turn is used for testing while the reminder is used for training. When there are insufficient significant pixels due to the lower center significance value, it is unnecessary to consider the other two measures. After the morphological closing operation followed by the opening one is applied to the significant pixels, a minimum bounding rectangle (MBR) for the biggest connected component of significant pixels is selected. If the size of the MBR in an image is less than 5% of the total image size, the image is

classified as a non-object image without additional process. We achieve an 84.2% classification accuracy on the total data. Table 1 shows classification accuracy based on the precision, recall and F-measure. From the precision and recall of the total data we can see that more object images are misclassified to non-object images class than the opposite. Table 2 shows the classification results for each measure. The measure on the edge strength at the object boundary provides the best accuracy.



**Fig. 3.** The distribution of values for 450 object images and 450 non-object images when each measure is applied to: (a) the center significance, (b) the variance of significant pixels, and (c) the edge strength measure at the object boundary

**Table 1.** Evaluation of the classification result by using 6-fold cross-validation

	Object	Non-Object
Precision	0.90	0.80
Recall	0.78	0.90
F-measure	0.83	0.85

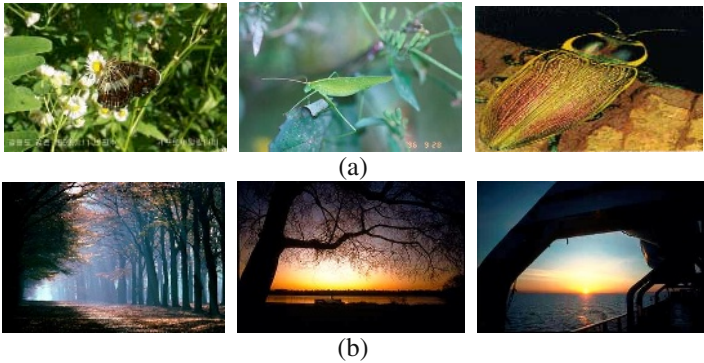
Fig. 4 shows a representative subset of the misclassified object and non-object images. The main reason for the misclassification of object images is the similarity of the object to the background in the color distribution. The reason for the misclassification of the non-object images is that center regions of the images are surrounded by regions with abrupt change in the color distribution.

The object images can be subdivided into images with objects of interest and otherwise. The objects of interest must be more useful in CBIR than the others. In this

**Table 2.** Evaluation of the classification results for each measure by using 6-fold cross-validation

		Center Significance	Variance of significant pixels	Edge Strength at Object Boundary
Object	Precision	0.72	0.76	0.75
	Recall	0.68	0.74	0.80
	F-measure	0.70	0.75	0.78
Non-Object	Precision	0.70	0.75	0.79
	Recall	0.73	0.77	0.73
	F-measure	0.71	0.76	0.76

paper, we re-conduct the classification in view of objects of interest and non-object. A human subject classifies 94.9% (427) of the object images into the images with objects of interest. The percentage of objects of interest can be changed depending on the set of images selected. We also compute the classification accuracy when the non-interesting object images are assigned to the non-object image class. Table 3 shows the classification accuracy in this case.



**Fig. 4.** A subset of the misclassified (a) object and (b) non-object images

**Table 3.** Evaluation of the classification result in view of the object of interest

	Object	Non-Object
Precision	0.88	0.82
Recall	0.79	0.90
F-measure	0.83	0.86

We have used a neural network as a classifier so far. The classification accuracy may depend on the classifier to be used. To compare with other classifiers, we also adopt other classifiers, Naïve Bayes [10], Decision Table [11], Decision Tree classifier [12]. Table 4 shows the classification accuracy according to the classifiers. The classification results show similar accuracies regardless to the classifiers.

**Table 4.** Evaluation of the classification results according to the classifiers

		Naïve Bayes	Decision Table	Decision Tree
Object	Precision	0.83	0.86	0.84
	Recall	0.84	0.80	0.81
	F-measure	0.83	0.83	0.83
Non-Object	Precision	0.83	0.82	0.82
	Recall	0.82	0.87	0.85
	F-measure	0.83	0.84	0.83

## 5 Conclusions

We proposed a classification method that classified the images into the object and non-object image classes with an 84.2% accuracy. For this classification we proposed three measures (the center significance, the variance of significant pixels, and the edge strength at the object boundary) and trained the neural network based on them. When we used other classifiers, Naïve Bayes, Decision Table, and Decision Tree, instead of a neural network, we obtained similar classification accuracy. Our work is applicable to improve the performance of the image retrieval and image indexing.

## References

1. Vailaya, A., Jain, A.K., and Zhang, H.J.: On Image Classification: City images vs. landscape. *Pattern Recognition*. **31**(12) (1998) 1921-1936
2. Szummer, M., and Picard, R.W.: Indoor-outdoor image classification. *IEEE Int'l Workshop Content-Based Access Image Video Databases*. (1998) 42-51
3. Vailaya, A., Figueiredo, M.A.T., Jain, A.K., and Zhang, H.J.: Image Classification for Content-Based Indexing. *IEEE Trans. on Image Processing*. **10**(1) (2001) 117-130
4. Kim, S.Y., Park, S.Y., and Kim, M.H.: Central Object Extraction for Object-Based Image Retrieval. *Int'l Conf. on Image and Video Retrieval (CIVR)*. (2003) 39-49
5. Serra, J.R. and Subirana, J.B.: Texture Frame Curves and Regions of Attention Using Adaptive Non-cartesian Networks. *Pattern Recognition*. **32** (1999) 503-515
6. Park, S.B., Lee, J.W., and Kim, S.K.: Content-based image classification using a neural network. *Pattern Recognition Letter*. **25** (2004) 287-300
7. Huang, J., Kumar, S.R., Mitra, M., Zhu, W.J., and Zabih, R.: Image Indexing Using Color Correlograms. *Proc. Computer Vision and Pattern Recognition*. (1997) 762-768
8. Lippmann, R.P.: An introduction to computing with neural nets. *IEEE ASSP Magazine*. (1994) 4-22
9. Witten, I.H., Frank, E.: *Data Mining*. Academic Press. (2000)
10. Good, I.J.: *The estimation of Probabilities: An Essay on Modern Bayesian Methods*. MIT Press, MA USA (1965)
11. Kohavi, R.: The Power of Decision Tables. *Proceedings of the European Conference on Machine Learning, Lecture Notes in Artificial intelligence 914*, Springer Verlag, Berlin Heidelberg NY (1995)
12. Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J.: *Classification and Regression Trees*. Wadsworth, Belmont (1984)

# Video Segmentation Using Hidden Markov Model with Multimodal Features

Tae Meon Bae, Sung Ho Jin, and Yong Man Ro

IVY Lab., Information and Communications University (ICU),  
119, Munjiro, Yuseong-gu, Deajeon, 305-714, Korea  
{heartles, wh966, yro}@icu.ac.kr

**Abstract.** In this paper, a video segmentation algorithm based on Hidden Markov Model classifier with multimodal feature is proposed. By using Hidden Markov Model classifier with both audio and visual features, erroneous shot boundary detection and over-segmentation were avoided compared with conventional algorithms. The experimental results show that the propose method is effective in shot detection.

## 1 Introduction

With the rapid increase of the multimedia consumption, automatic or semi-automatic video authoring and retrieval systems are inevitable to effectively manipulate a large amount of multimedia data. Because video segmentation is basic operation in authoring and retrieving video contents, it is important to detect precise shot boundaries and segment a video into semantically homogeneous units for high-level video processing.

In conventional video segmentation algorithms, the detection of camera shot changes mostly had been focused on. To detect shot changes, visual clues such as luminance, color histogram differences of neighboring frames had been utilized [1-3]. But the conventional methods have the following problems.

First, the visual feature difference between neighboring frames is too small to detect gradually changed shots. To resolve this problem, the approach calculating the difference between every  $n$ -th frame was proposed [4-6]. The difference value could be amplified enough to detect shot change by the certain level of a threshold value. However, noises due to motion or bright variation are also amplified, thereby video is over-segmented. And the position of gradual shot boundary is not exact in this approach.

Second, an empirically obtained threshold value is generally used to detect shot boundaries. But the value of the feature difference between frames depends on the characteristics of the video content. So, shot detection methods with the constant threshold value could cause the miss of shot boundaries or over-segmentation. John S. Boreczky and Lynn D. Wilcox[7] suggested an HMM(Hidden Markov Model) based approach to avoid problems rising from the fixed threshold value. In their work, camera effects are described as the state of the HMM. The frames that belong to a 'shot' state are grouped into a video segment. The luminance difference, motion

vectors, and audio cepstral vector difference are used as feature values. Because the occurrences of camera effects and shot transitions in the training video determine the state transition probability, its performance may depend on the genre of the input video. They had used various genre video contents as training data.

Third, even though the shot boundary is correctly detected, the video may over-segment in the semantic point of view. It is possible to merge dependent shots into a meaningful scene by audio information. Though audio features are not superior to visual ones in finding shot boundary, they could give meaningful information when visual features could not figure out the scene boundary, or when there is semantic similarity in audio signal between video segments.

In the previous HMM based approach [7], the temporal characteristics of camera effect and video segment are not considered since each camera effect and video segment are modeled by one state. The detection of the each effect mainly depends on the output probability of the state that belongs to the effect and state transition probabilities that represent statistical occurrences of camera effects.

Other HMM based approaches focused on the modeling the semantic scenes [8], where every shot should have its own semantics. These approaches can be applied after video segmentation to detect or merge shot by semantics.

In this paper, we propose new HMM based framework. In the proposed method, camera effects and video segment have their own HMMs, where the probabilistic characteristics of the effects can be represented in temporal direction. Furthermore, we employed the method to emphasize the temporal characteristics of the abrupt and gradual shot change [4]. And audio features were used to detect shot boundaries in cooperation with visual features to enhance semantic video segmentation.

The multimodal features in the proposed method employs Scalable Color, Homogeneous Texture, Camera Motion, and Audio Spectrum Envelop descriptor in MPEG-7 audio/visual features, which could provides audio and visual descriptors that represent the characteristics of the multimedia contents [9-13]. These descriptors make it possible to separate video content manipulation from feature extraction. Namely, video authoring or retrieval tool can use metadata which contain pre-extracted feature sets without accessing raw video data in the video content database.

## 2 Video Segmentation Using HMM Classifier

Figure 1 shows the procedure of the proposed video segmentation. Hidden Markov Model is adopted in video segmentation. First, MPEG-7 audio and visual features are extracted from video content. Using these features, HMM classifies video frames into non-transition region, abrupt transition region, and gradual transition region. Shot boundary detection could be considered as detection of abrupt or gradual transition regions. After that, falsely detected shot boundaries are rejected by shot transition verification process. Audio feature is a clue of video segmentation as well. It could improve semantic similarity in the segmentation. It can be assumed that the similar audio characteristics have the similar semantics.

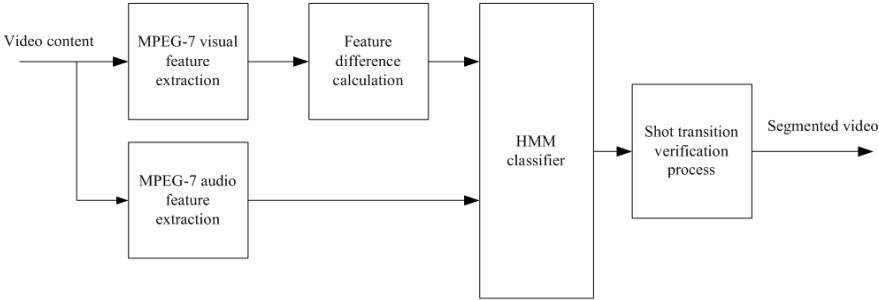


Fig. 1. Proposed video segmentation process

2.1 HMM Based Video Segmentation

The objective of the proposed HMM classier is to detect meaningful shot boundaries. The visual feature difference between every  $K$ -th frame shows a specific waveform in temporal domain as shown in Fig. 2. The shape of an abrupt shot change signal is similar to rectangular while the shape of gradual change is similar to triangular. We trained HMM to recognize these waveforms. Using HMM, shot boundaries can be modeled into a set of features and specific waveforms, same as speech recognition system [14]. For an abrupt shot boundary, the HMM could detect rectangular waveform with the length of  $K$ . The HMM can also detect gradual boundaries that is arbitrary in length by using Viterbi algorithm which has the dynamic time warping characteristics [14].

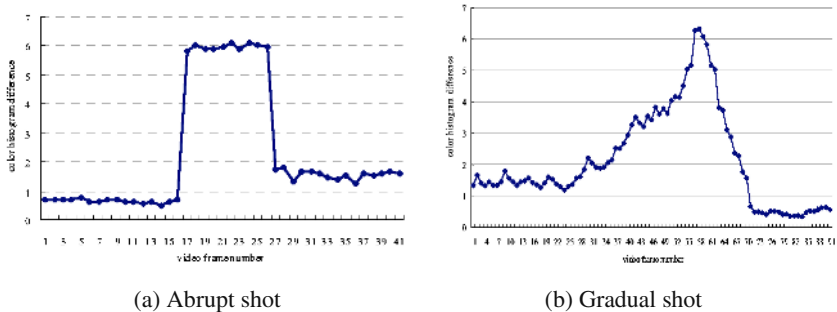
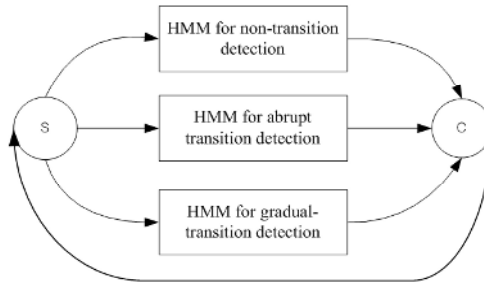


Fig. 2. The difference feature values of Scalable Color descriptor between every 10-th frames for an example video

Figure 3 shows the structure of the HMM classifier. Three HMMs are designed to classify video segments into non-transition, abrupt transition, and gradual transition region. Here, ‘S’ is starting state, and ‘C’ is collector state. Ergodic state transition model is used in the HMM.

Because states in the HMM are not defined heuristically, they are hidden and unknown, which makes hard to realize HMM classifier system. We should know the number of states for each HMM and training data matched for each state. These problems are solved by the iterative method. First, the training data are clustered according to the initial number of states and then the HMM is trained using the clustered data. Changing the number of state, the performance of the HMM is measured. And the number of states that shows the best performance is selected. We used Expectation Maximization (EM) algorithm as a clustering method, in which output probability is assumed to be joint Gaussian distribution.

Scalable Color, Homogeneous Texture, Edge histogram, and Audio Spectrum Envelop features are used to represent the states. Scalable color histogram is popularly used to measure the similarity between images in image retrieval [2]. And Edge Histogram detects spatial similarity between images, and detects camera break that could not be detected by Scalable color. But Edge Histogram is space variant. We use Homogeneous Texture feature which has space invariant characteristics [15]. To differentiate abrupt cut and gradual change, scalable color difference between neighboring frames are also used as feature data.



**Fig. 3.** HMM classifier for shot boundary detection

AudioSpectrumEnvelop descriptor has 10 coefficients, and among them, we use the value of the lowest frequency band. The neighboring non-transition regions are assumed to be semantically similar if an audio signal exists in the transition region (shot boundary region). Figure 4 shows the extracted audio feature and camera cuts. As seen, the first 3 shots are semantically similar even though they are separated by camera cut. The audio frame length is 18.75 milliseconds and the hop size is 6.25 milliseconds. To synchronize with video features, we averaged audio features according to the video frame rate.

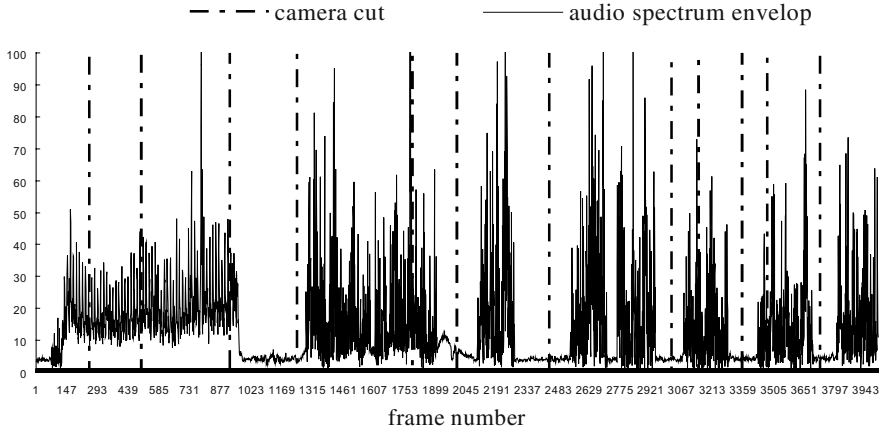
The feature vector is composed of  $DT_k(n)$ ,  $DS_l(n)$ ,  $DE_k(n)$ ,  $DS_k(n)$ , and  $ASE(n)$ ; where  $DS_l(n)$  is L1 norms of difference between  $n$ -th and  $(n+1)$ -th frames of Scalable Color;  $DT_k(n)$ ,  $DE_k(n)$ ,  $DS_k(n)$  are L1 norm of difference between  $n$ -th and  $(n+k)$ -th frames of Homogeneous Texture, Edge Histogram, and Scalable Color descriptor, respectively;  $ASE(n)$  is averaged spectrum value of the AudioSpectrumEnvelop feature. To retain the mean and variance of the features,  $DT_k(n)$ ,  $DS_l(n)$ ,  $DE_k(n)$ ,  $DS_k(n)$ , and  $ASE(n)$  are normalized as follows



$$f_{norm}(n) = \frac{f(n)}{\sqrt{\sum_{n=0}^{N-1} f^2(n)}} \quad (1)$$

where  $f(n)$  is  $n$ -th frame feature,  $f_{norm}(n)$  is normalized feature, and  $N$  is the number of frames in the video clip. Therefore, input feature vector for HMM can be written as

$$\vec{F}_{norm}(n) = (DT_{Knorm}(n), DS_{Inorm}(n), DS_{Knorm}(n), DE_{Knorm}(n), ASE_{norm}(n)) \quad (2)$$



**Fig. 4.** The envelop of audio spectrum (0~88.338Hz): “science eye” video clip

## 2.2 Verification of Detected Shot Transition

After detecting abrupt and gradual shot transition regions, erroneously detected shots are rejected by verification process proposed in this paper. We observed two main causes for erroneous shot transition detection.

First, true shot transition regions do not contain camera motions. Shot transition regions are rejected when camera motions are detected within transition region. Equation (3) shows a flag indicating the rejection by camera motion detection.

$$reject\_by\_motion(T) = \begin{cases} true, & \text{if } cm > 0 \\ false, & \text{otherwise} \end{cases}, \quad (3)$$

where  $cm = \sum_{i=0}^{13} camera\_motion\_practionalPresence[i]$

Here,  $T$  represents shot transition region, and  $\text{camera\_motion\_practionalPresence}[i]$  represents  $i$ -th camera motion defined in MPEG-7 camera motion descriptor. 14 kinds of camera motions except the fixed motion are used in detecting camera motion.

Secondly, false shot transition detection could be caused by short-term disturbance such as dominant object movement or bright variation. In this false shot transition, the length of the transition region detected by HMM is always shorter than the length of abrupt shot transition region,  $K$ . So rejection flag can be written as,

$$\text{reject\_by\_length}(T) = \begin{cases} \text{true}, & \text{if } T.\text{length} < K \\ \text{false}, & \text{otherwise} \end{cases}, \quad (4)$$

where  $T.\text{length}$  represents the length of detected region,  $T$ . If Eq. (3) or (4) returns true value, the transition region  $T$  is re-labeled as non-transition region.

Verification process by camera motion can be included in the HMM by using camera motion as HMM input feature. But there are several problems in this approach. First, camera motion is shot level feature, so we should know the start and end frame to calculate camera motion. Without knowing the shot boundary, camera motion estimation results erroneous value. Second, to use camera motion as input feature of HMM, it is need to extract camera motion frame by frame, which is very time consuming task.

### 3 Experiment

In the experiment, we tested the performance and robustness of the proposed method. The performance was compared with the threshold based method. And robustness was tested by applying the proposed method to diverse genre videos. Effectiveness of audio feature was also examined.

#### 3.1 Experimental Condition

In experiments, we used MPEG-7 video database [16]. We tested three different video genres, *e.g.*, news, golf, and drama. Table 1 shows number of shot boundaries in the tested video set used in the experiments. The performance of the abrupt shot boundary and the gradual shot boundary detection were tested with the video set in Table 1.

The HMMs were trained using 10 minutes video contents. Mono channel, 16 KHz audio was used to extract audio features. Six states for non-transition, one state for abrupt transition, and two states for gradual transition HMM were obtained by iterative performance evaluation in training HMM. The frame difference  $K$  is set to 10 in the experiment, which is empirically determined.

#### 3.2 Experimental Results

Table 2, 3, and 4 show the results of shot boundary detection by the proposed method. Because there are several different genres in the news clip, the experimentation using

**Table 1.** Number of shot boundaries in the tested MPEG-7 video set

Test video	Item Number	Content	Number of frame	Shot boundary	
				Abrupt	Gradual
Golf.mpg	V17 NO. 26	Golf	17909	31	17
Jornaldanoite1.mpg	V1 NO. 14	News	33752	184	36
Pepa_y_Pepe.mpg	V7 NO. 21	Drama	16321	20	1

this clip is a good way to test the robustness of the segmentation algorithm. The news clip used for the experiment contains soccer, marathon, dance, and interview scenes.

As seen in Table 2, there are little missed abrupt shot boundaries. Most missed abrupt cuts were due to aliasing effect of adjacent gradual shots. When abrupt shots are occurred near the gradual shot, the two regions are likely to be overlapped and converged one shot. The missed gradual shots in Table 2 were related with power of signal. In the news clip, there were burst misses of 5 dissolves occurred from 32670 to 33003-th frame, where the video frames neighboring the dissolve regions are similar.

**Table 2.** Results of abrupt shot detection

Contents	Abrupt Shot boundary		
	Original	Correct	Missed
News	184	177	7
Golf	31	31	0
Drama	20	18	2

**Table 3.** Results of gradual shot detection

Contents	Gradual shot boundary		
	Original	Correct	Missed
News	36	31	5
Golf	17	15	2
Drama	1	1	0

Table 4 and 5 shows the effects by the audio feature. In HMM, visual only feature could be better than multimodal HMM in terms of the correctness of shot boundary detection. However, in the news clip as seen in Table 4 and 5, 6 additional missed gradual shot boundaries are semantically merged by audio feature, which means audio feature could lead to merge semantically homogeneous shots. For example, an anchor speech could be continued even though there exist camera cuts. These shots are semantically linked and audio feature could combine them into the same scene. The one additional missed shot in drama clip is also due to audio similarity with neighboring shot. The HMM classifier without audio feature could detect camera breaks or cuts more correctly, but fail to merge shot-fragments into meaningful shot.

We compared the proposed method with the threshold based algorithm that shows good result in detecting gradual shot boundary [6]. In this algorithm, color histogram difference of every 20-th frame is calculated. And shot boundary is determined when the ratio of difference is larger than predefined threshold value that is selected enough to detect gradual shot boundary. From Table 4 and 6, the proposed method and

threshold-based method are almost same in the number of correctly detected shot boundary. But, for the number of falsely detected shot boundary, the proposed method is smaller than that of the threshold based method. It means the proposed method could prevent over-segmentation effectively.

**Table 4.** Results of shot detection with audio

Contents	Total Shot boundary				Accuracy	
	Original	Correct	Missed	False	Recall	Precision
News	220	208	12	17	0.945	0.924
Golf	48	46	2	22	0.958	0.676
Drama	21	19	2	4	0.904	0.826

**Table 5.** Results of shot detection without audio feature

Contents	Total number of shot boundary				Accuracy	
	Original	Correct	Missed	False	Recall	Precision
News	220	214	6	17	0.973	0.926
Golf	48	46	2	22	0.958	0.676
Drama	21	20	1	5	0.952	0.800

**Table 6.** Results of shot detection by color histogram thresholding

Contents	Total number of shot boundary				Accuracy	
	Original	Correct	Missed	False	Recall	Precision
News	220	205	15	22	0.932	0.903
Golf	48	46	2	50	0.900	0.479
Drama	21	21	0	27	1.000	0.438

Table 7 shows the performance of verification process. It reduced 50 to 60% of falsely detected shots. Camera motion and short term disturbance were dominant reasons of false detection in the analysis with the verification process. Some of the falsely detected shots were due to the failure of camera motion estimation.

**Table 7.** Number of flase shots related wit the verification process

Contents	False before the processing	False after the processing
News	30	17
Golf	40	22
Drama	4	4

## 4 Conclusions and Future Works

In this paper, a video segmentation algorithm with multimodal feature is proposed. By using both audio and visual features, erroneous shot boundary detection and over-segmentation were avoided compared with conventional algorithms. The experimental results show that the propose method is effective in shot detection. Also

the results showed that shots could be merged into more meaningful scenes. To increase merging efficiency, shot level features could be considered as a future work.

**Acknowledgement.** This research was supported in part by “SmarTV” project from Electronics and Telecommunications Research Institute.

## References

1. Wang, H., DivaKaran, A., Vetro, A., Chang, S.F., Sun, H.: Survey of Compressed-domain features used in audio-visual indexing and analysis, *J. Vis. Commun. Image R.* 14, (2003) 150-183
2. Gargi, U., Kasturi, R., Strayer, S.H.: Performance Characterization of Video-Shot-Change Detection Methods, *IEEE Trans. On CSVT*, Vol.10, (2000)
3. Huang, J., Liu, Z., Wang, Y.: Integration of audio and visual information for content-based video segmentation, in *Proc. IEEE Int. Conf. Image Processing (ICIP98)*, vol. 3, Chicago, IL, (1998) 526-530
4. Zhang H.J., Kankanhalli, A., Smoliar, S.W.: Automatic partitioning of full-motion video, *Multimedia System*, (1993) 10-28
5. Li, Y., Kuo, C.C.J.: *Video Content Analysis Using Multimodal Information*, Kluwer Academic publisher, (2003)
6. Shim S.H., et al.: Real-time Shot Boundary Detection for Digital Video Camera using The MPEG-7 Descriptor, *SPIE Electronic Imaging*, Vol. 4666, (2002) 161-171
7. Boreczky, J.S., Wilcox, L.D.: A hidden Markov model framework for video segmentation using audio and image features, in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-98)*, Vol. 6, Seattle, WA, May (1998) 3741-3744
8. Wolf, W.: Hidden Markov Model Parsing of Video Programs, in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-97)*, Vol. 4, April. (1997) 2609-2611
9. ISO/IEC JTC1/SC29/WG11 N4980, MPEG-7 Overview, July (2002)
10. ISO/IEC 15938-4, Information Technology – Multimedia Content Description Interface – Part 4: Audio, (2001)
11. ISO/IEC 15938-3, Information Technology – Multimedia Content Description Interface – Part 3: Visual, July (2002)
12. ISO/IEC JTC1/SC29/WG11MPEG02/N4582, MPEG-7 Visual part of eXperimentation Model Version 13.0, March (2002)
13. Manjunath, Salembier, P., Sikora, T.: *Introduction to MPEG-7 Multimedia Content Description Interface*, B.S., John wiley & Sons, LTD, (2002)
14. Huang, X., Acero, A., Hon, H.W.: *Spoken Language processing*, Prentice Hall, (2001) 377-409
15. Ro, Y.M., et al.: MPEG-7 Homogeneous Texture Descriptor, *ETRI Journal*, Vol. 23, Num. 2, June (2001)
16. ISO/IEC JTC1/SC29/WG11/N2467, Description of MPEG-7 Content Set, Oct. (1998)

# Feature Based Cut Detection with Automatic Threshold Selection

Anthony Whitehead<sup>1</sup>, Prosenjit Bose<sup>1</sup>, and Robert Laganier<sup>2</sup>

<sup>1</sup> Carleton University, School Computer Science,  
Ottawa, Ontario, Canada  
{awhitehe, jit}@scs.carleton.ca

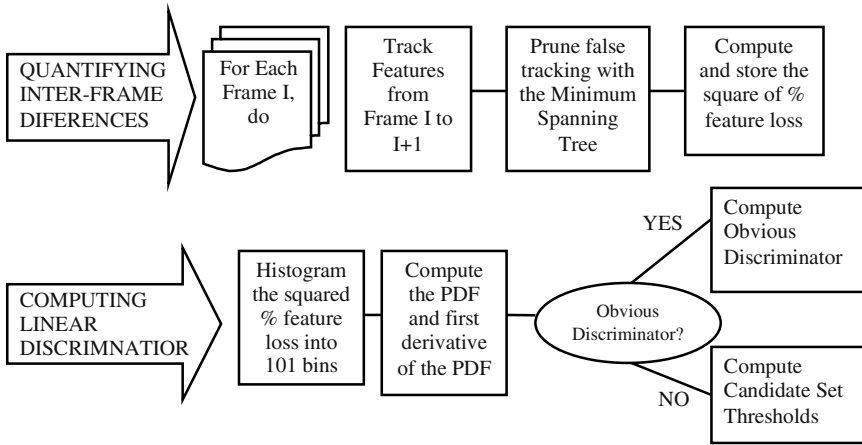
<sup>2</sup> University of Ottawa, School of Information Technology and Engineering  
Ottawa, Ontario, Canada  
laganier@site.uottawa.ca

**Abstract.** There has been much work concentrated on creating accurate shot boundary detection algorithms in recent years. However a truly accurate method of cut detection still eludes researchers in general. In this work we present a scheme based on stable feature tracking for inter frame differencing. Furthermore, we present a method to stabilize the differences and automatically detect a global threshold to achieve a high detection rate. We compare our scheme against other cut detection techniques on a variety of data sources that have been specifically selected because of the difficulties they present due to quick motion, highly edited sequences and computer-generated effects.

## 1 Introduction

In 1965, Seyler developed a frame difference encoding technique for television signals [1]. The technique is based on the fact that only a few elements of any picture change in amplitude in consecutive frames. Since then much research has been devoted to video segmentation techniques based on the ideas of Seyler. Cut detection is seemingly easily solved by an elementary statistical examination of inter-frame characteristics; however a truly accurate and generalized cut detection algorithm still eludes researchers. Reliable shot boundary detection forms the cornerstone for video segmentation applications as shots are considered to be the elementary building blocks that form complete video sequences. Applications such as video abstraction, video retrieval and higher contextual segmentation all presuppose an accurate solution to the shot boundary detection problem [2,3,4,5,6]. Automatic recovery of these shot boundaries is an imperative primary step, and accuracy is essential.

A hard cut produces a visual discontinuity in the video sequence. Existing hard cut detection algorithms differ in the feature(s) used to measure the inter-frame differences and in the classification technique used to determine whether or not a discontinuity has occurred. However, they almost all define hard cuts as isolated peaks in the features time series. In [7, 10] complete surveys are given on techniques to compute inter-frame differences and classify the types of transition. A variety of metrics have been suggested to work on either raw or compressed video forming the basis of our comparisons. Figure 1 outlines our proposed method.



**Fig. 1.** Diagram of proposed system to compute cuts

This paper is structured as follows: Section 2 details our method for quantifying inter-frame differences by using a stable feature tracking mechanism. Section 3 details our method of automatic threshold selection by examining the density properties of the inter-frame differences. Section 4 performs a variety of experiments. Section 5 outlines areas that present difficulties for the proposed method; we summarize the results and draw some conclusions. In this work we concentrate on the detection of cuts as they represent instantaneous frame pair changes in time. The method is easily expandable to allow inferences over several frames in time. I.e. Computing the displacement vectors of tracked features yields object and camera motion information.

## 2 Quantifying Interframe Differences

We propose a new approach that uses feature tracking as a metric for dissimilarity. Furthermore we propose a methodology to automatically determine a threshold value by performing density estimation on the squared per-frame lost feature count. It has been reported that the core problem with all motion-based features used to detect cuts is due to the fact that reliable motion estimation is far more difficult than detecting visual discontinuity, and thus less reliable [7]. Effectively, a simple differencing technique is replaced with a more complex one. Experimentally we have found that the proposed feature tracking method performs flawlessly on all simple<sup>1</sup> examples where pixel and histogram based methods did not achieve such perfect results. We continue by outlining the feature tracking method, a pruning algorithm and a signal separation methodology. We follow up in the next section with a method to dynamically select a global threshold. Each block within Figure 1 is detailed in this section and the next.

<sup>1</sup> Here we define simple to be cases of clearly obvious cuts, which were well separated over time and space.

## 2.1 Feature Tracking

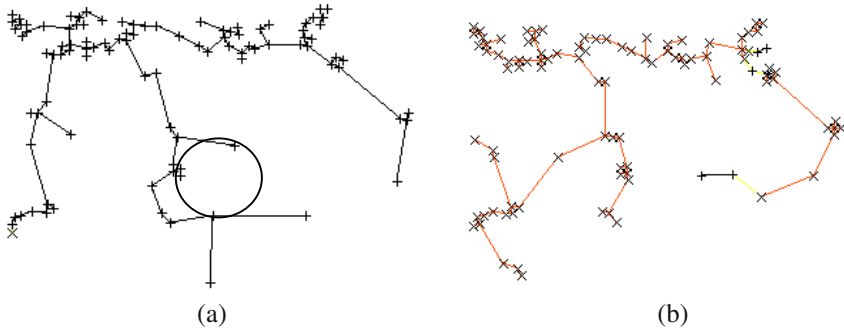
Previous feature based algorithms [8,9] rely on coarse-grained features such as edges and do not track edge locations from frame to frame. Rather they rely on sufficient overlap of a dilated edge map and search a very small local area around the original edge locations. In contrast, the proposed method of tracking fine-grained features (corners and texture) on a frame-by-frame basis using a pyramid approach is less constrained by the original feature location. Furthermore, the edge based method could only achieve frame rates of 2 frames per second [10], while our proposed method achieves over 10 frames per second on standard video dimensions. These reasons allow our proposed method to be more robust to object and camera motions yet remain practical. Cuts are detected by examining the number of features successfully tracked in adjacent frames, refreshing the feature list for each comparison.

The feature tracker we use is based on the work of Lucas and Kanade [11]. Space constraints prevent full disclosure of the work in [11], but briefly, features are located by examining the minimum eigenvalues of a  $2 \times 2$  image gradient matrix. The features are tracked using a Newton-Raphson method of minimizing the difference between the two windows around the feature points. Due to the close proximity of frames in video sequences, there is no need to perform affine window warping, which greatly reduces the running time requirements. The displacement vector is computed using a pyramid of resolutions because processing a high resolution image is computationally intense. A multi-resolution pyramid within the feature tracker reduces the resolution of the entire image and tracking occurs by locating a features general area in the lowest resolution and upgrading the search for the exact location as it progresses up the pyramid to the highest resolution.

## 2.2 Pruning False Tracking

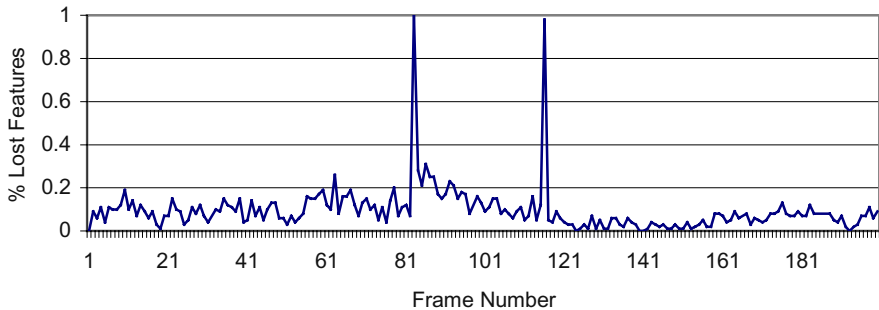
In the case of a cut at frame  $i$ , all features should be lost between frames  $i$  and  $i+1$ . However, there are cases where the pixel areas in the new frame coincidentally match features that were being tracked. In order to prune these coincidental matches, we examine the minimum spanning tree of the tracked and lost feature sets. We can see from Figure 2 (b), in the case of a cut, that a very small percentage of features are tracked. These tracked features are, in fact, erroneous. We can remove some of these erroneous matches by examining properties of the minimum spanning tree (MST) of the tracked and lost feature sets. By severing edges that link tracked features to lost features we end up with several disconnected components within the graph (MST). Any node in the graph that becomes a singleton has its status changed from tracked to lost, and is subsequently included in the lost feature count. The property we are exploiting here the fact that erroneously tracked features will be minimal and surrounded by lost features. Clusters of tracked (or lost) features therefore have localized support that we use to lend weight to our assessment of erroneous tracking.





**Fig. 2.** Minimum spanning trees for two consecutive frames (+) are tracked features, (X) are lost features. (a) high proportion of successfully tracked features from previous frame (b) features cannot be found in high proportion, indicating a cut. The circle shows erroneous tracking.

Our inter-frame difference metric is the percentage of lost features from frames  $i$  to  $i+1$ . This corresponds to changes in the minimum spanning tree, but is computationally efficient. Because we are looking to automatically define a linear discriminator between the cut set and the non-cut set, it is advantageous to separate these point sets as much as possible. In order to further separate the cut set from the non-cut set, we square the percent feature loss which falls in the range  $[0..1]$ . This has a beneficial property of ensuring the densities of the cut set and the non-cut set are further separated and thus ease the computation of a discriminating threshold. The idea here is that in the case of optimal feature tracking, non-cut frame pairs score 1 (all features tracked) and cut frame pairs score 0, no features tracked. Squaring, in the optimal case, has no effect as we are already maximally separated. However, in practice, squaring forces the normalized values for non-cut frame pairs closer to zero. Figure 3 shows this stretching of the inter-frame differences and how cuts and non-cuts are well separated. Now that we have determined interframe differences, we continue by discussing the classifier.



**Fig. 3.** Cuts expose themselves as high inter-frame feature loss.

### 3 Automatically Determining a Linear Discriminator

There is no common threshold that works for all types of video. Having a difference metric and a method to further separate the cut set from the non-cut set, we can now compute the linear discriminator for the two sets. There are two classes of frame differences, cuts and non-cuts; and our goal is to find the best linear discriminator that maximizes the overall accuracy of the system.

The cut set and the non-cut set can be considered to be two separate distributions that should not overlap, however, in practice they often do. When the two distributions overlap, a single threshold will result in false positives and false negatives. An optimal differencing metric would ensure that these two distributions do not overlap; in such a case the discriminating function is obvious and accuracy is perfect. Figure 4 demonstrates this. The quality of the difference metric directly affects the degree to which the two distributions overlap, if any; and until an optimal difference metric is proposed, the problem of optimal determination of the discriminator must be considered.

We have opted to examine the density of the squared inter-frame difference values for an entire sequence. The idea here is that there should be two distinct high-density areas, those where tracking succeeded (Low feature loss) and those where tracking failed (high feature loss). In practice, this situation appeared about 50% of the time in our data set. We will introduce the idea of a candidate set in section 3.2, which is the set of features that can be discriminated by zero crossings of the probability density function that characterizes the densities of the inter-frame differences. It needs to be noted here that while we examine the density for the entire sequence to determine a global threshold, it is possible to apply the method outlined next in a windowed manner to determine localized thresholds.

#### 3.1 Density Estimation

In order to auto-select a threshold, we examine the frequency of high and low feature loss. We are looking to exploit the fact that the ratio of non-cuts to cuts will be high, and therefore the ratio of low feature loss frame pairs to high feature loss frame pairs will also be high. As the frame to frame tracking of features is independent of all other video frames, we have  $n$  independent observations from an  $n+1$  frame video sequence. The extrema of the probability density function (PDF) can be used to determine the threshold to use. We can use the statistical foundations of density estimation to estimate this function.

The kernel density estimator for the estimation of the density value  $f(x)$  at point  $x$  is defined as

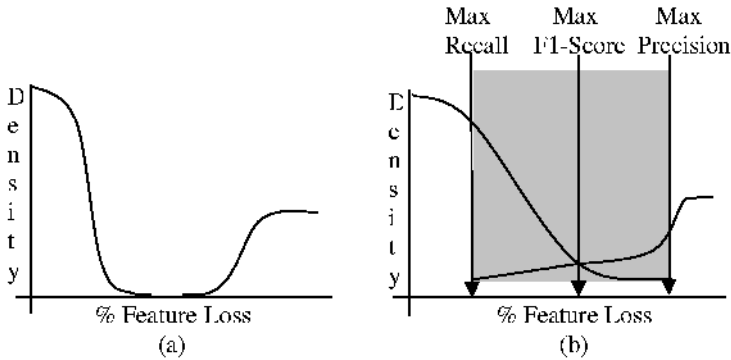
$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x_i - x}{h}\right) \quad (1)$$

Where  $K(\bullet)$  is the so-called kernel function,  $h$  is the window size, and  $n$  is the number of frames. Through a series of experiments, the triangular kernel ( $K(\alpha) = 1 - |\alpha|$ ) was selected because it did not over-smooth locally, making the determination of extrema

easiest. Kernel widths of 7 have provided good results in our experiments. The PDF can be estimated in linear time because we have a small histogram with only 101 bins.

### 3.2 The Candidate Sets

Non-overlapping sets of distributions are very easily determined by looking for a large plateau of zero density as in Figure 4(a). The first appearance of a large plateau of zero density indicates the range of the separation point. Selecting the extreme end point (closest to the cut set) for the threshold has yielded the correct result on all cases of non-overlapping distributions in our test suite. In practice, the problem is not so simple. We next introduce the ideas around what we term the candidate sets.



**Fig. 4.** (a) Non-overlapping distributions of cuts/non-cuts, error free discrimination (b) Overlapping distribution of cuts and non-cuts. Error region outlined in gray, and maximum Recall, Precision and F1 score thresholds pointed out.

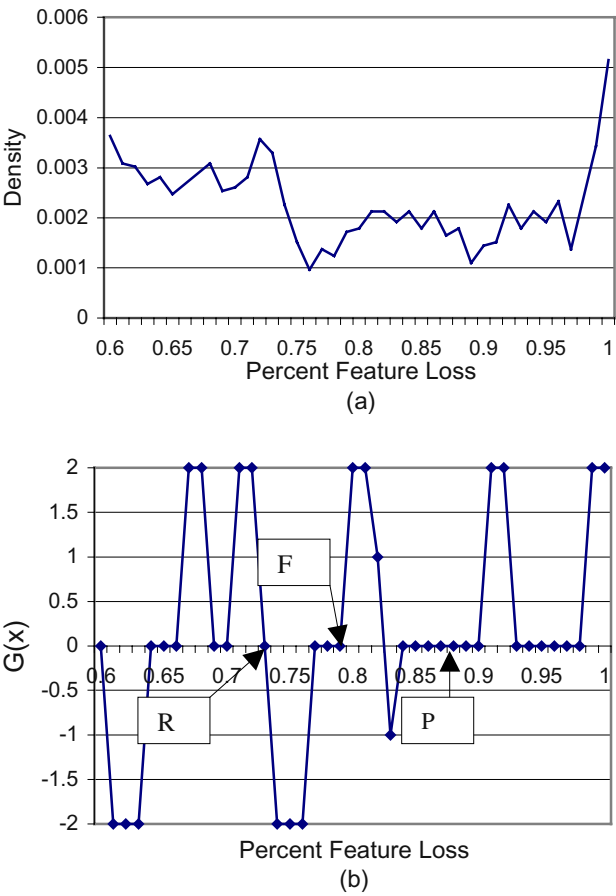
We define 3 candidate sets, where each set contains the frames that maximize the precision, F1 and recall rates. Precision is the portion of the declared cuts that were correct and is maximized when the non-cut distribution ends. Recall is portion of the cuts that were declared correctly and is maximized when the cut distribution ends. The F1 is a combination of precision and recall and is maximized at the intersection of the two distributions. Figure 4(b) shows the overlapping distributions and the position of the candidate set thresholds. Depending on user need, precision, recall or best overall performance (F1), thresholds for these candidate sets can be determined. The candidate sets are the 3 thresholds that for convenience we will call the precision set (P), the F1 set (F) and the recall set (R). The candidate sets are determined by examining zero crossings of the first derivative of the computed probability density function. There are often many consecutive zero crossings of the function over time, so we use a modified function  $G(x)$  to make the large changes in density more apparent.  $G(x)$  is defined using the following rules:

$$\begin{aligned}
 G(x) &= g(x) + g(x+1) : \\
 &\text{if } \hat{f}'(x) \leq 0 \text{ then } g(x) = 0 \\
 &\text{if } \hat{f}'(x) > 0 \text{ then } g(x) = 1
 \end{aligned} \tag{2}$$

The zero crossings are determined by starting from 100 percent feature loss and examining  $G(x)$  as  $x$  (% feature loss) decreases.

- (P) is the first zero crossing
- (F) The position of the minimum of PDF corresponding to the plateau of  $G(x)$  given:
  - If the next zero crossing has opposing direction as the first and is part of the plateau of first zero crossing use this plateau (i.e. is u or n shaped), otherwise use the next plateau.
- (R) The next subsequent zero crossing

The arrows in Figure 5(b) point to the zero crossings computed using the rules. The first zero crossing is at 98 (P) and because the next zero crossing at 93 is also an upwards direction (u shaped), we skip to the next plateau to determine F. The next zero crossing (not on the plateau) is used for R.



**Fig. 5.** (a) Original PDF (b) Modified first derivative ( $G(x)$ )

4 Experimental Results

In the experiment that follows, a selection of video clips that represent a variety of different video genres are used. The data set was specifically selected based on characteristics that cause difficulties with known methods. In particular, we selected clips with fast moving objects, camera motions, highly edited and various digitization qualities and lighting conditions. We compare the results of the proposed method against a pixel-based method with relative localization information and a histogram based method [12]. For the proposed method, we ran each sample through the system once and computed the F1 candidate set threshold as outlined in Section 4. For the two comparison methods, we preformed binary search to find the maximum F1 score.

Table 1. Results on data set.

	Proposed feature tracking method			Pixel Based method with localization			Histogram MethodCut Det (MOCA)		
Data Source	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
A	1	1	1	1	1	1	1	1	1
B	1	1	1	.825	.825	.825	1	.375	.545
C	.595	.870	.707	.764	.778	.771	.936	.536	.682
D	1	1	1	1	1	1	1	.941	.969
E	.938	1	.968	.867	.867	.867	.955	.700	.808
F	1	1	1	0	0	0	1	1	1
G	.810	.944	.872	.708	.994	.809	1	.667	.800
H	.895	.895	.895	.927	1	.962	.971	.895	.932
I	1	1	1	1	1	1	1	.500	.667
J	.497	.897	.637	.623	.540	.591	.850	.395	.540
AVG	.874	.961	.908	.774	.800	.783	.971	.701	.794
VAR	.034	.003	.018	.090	.101	.093	.002	.060	.036
DEV	.185	.054	.134	.301	.318	.304	.048	.246	.190

In Table 1, we present the results of running the 3 methods on the dataset. The proposed method outperforms both the histogram-based method and the pixel based methods. In most cases (8 of 10) the proposed method provides the best achievable F1 score. A simple statistical analysis of the overall capabilities is given at the end of Table 1. The average, variance and standard deviation for the 10 samples were computed. On average, the proposed method significantly outperforms the other two methods. The variance and the standard deviation show that the results offered by the proposed method are also more stable across a variety of different video genre. It is not surprising that ‘cutdet’ out performs the proposed system in H, because the abstract was created by the MOCA project, however it is surprising that the pixel based method outperformed both. In examples C and J, the F1 score was not maximized as the heuristic to determine the F1 candidate set threshold did not achieve the best

value, rather a good value. Within the range of the F1 candidate set threshold plateau, maximum F1 was achievable. For all the experiments we tracked 100 features with a minimum feature disparity of five pixels. The processing time for a frame pair is slightly less than 70 ms on a 2.2 GHz Intel processor on frames sized 320x240 pixels.

## 5 Conclusions

We have presented a feature-based method for video segmentation, specifically cut detection. By utilizing feature tracking and an automatic threshold computation technique, we were able to achieve F1, recall and precision rates that generally match or exceed current methods for detecting cuts. The method provides significant improvement in speed over other feature-based methods and significant improvement in classification capabilities over other methods. The application of feature tracking to video segmentation is a novel approach to detecting cuts. We have also introduced the concept of candidate sets that allow the user to prejudice the system towards results meeting their individual needs. This kind of thresholding is a novel approach to handling the overlapping region of two distributions, namely the cut set and the non-cut set in video segmentation. Space constraints prevented the complete description and further information is available in a full and complete technical report. Please contact the authors to access the technical report.

## References

1. A. Seyler: Probability distribution of television frame difference, Proc. Institute of Radio Electronic Engineers of Australia 26(11), pp 355-366, 1965
2. J. Lee and B. Dickinson: Multiresolution video indexing for subband coded video databases, in Proc. Conference on Storage and Retrieval for Image and Video Databases, 1994.
3. R. Lienhart: Dynamic Video Summarization of Home Video, SPIE Storage and Retrieval for Media Databases 2000., 3972, pp. 378-389, Jan. 2000.
4. R. Lienhart, S. Pfeiffer, and W. Effelsberg: Video Abstracting. Communications of the ACM, Vol. 40, No. 12, pp. 55-62, Dec. 1997.
5. M. M. Yeung and B.-L. Yeo: Video Visualization for Compact Presentation and Fast Browsing of Pictorial Content. IEEE Trans. on Circuits and Systems for Video Technology, Vol.7, No. 5, pp. 771-785, Oct. 1997.
6. A. Hampapur, R. Jain, and T. E. Weymouth: Production Model Based Digital Video Segmentation. Multimedia Tools and Applications, Vol.1, pp. 9-45, 1995.
7. R. Lienhart: Reliable Transition Detection In Videos: A Survey and Practitioner's Guide. International Journal of Image and Graphics (IJIG), Vol. 1, No. 3, pp. 469-486, 2001.
8. R Zabih, J. Miller, and K. Mai: A Feature-Based Algorithm for Detecting and Classifying Scene Breaks, Proc. ACM Multimedia, pp. 189-200, 1995
9. R Zabih, J. Miller, and K. Mai: A Feature Based Algorithm for detecting and Classifying Production Effects, Multimedia Systems, Vol 7, p 119-128, 1999.
10. A Smeaton et al.: An Evaluation of Alternative Techniques for Automatic Detection of Shot Boundaries in Digital Video, in Proc. Irish Machine Vision and Image Processing, 1999.
11. B. Lucas and T. Kanade: An Iterative Image Registration Technique with an Application to Stereo Vision, Int. Joint Conference On Artificial Intelligence. pp 674-679, 1981.
12. S. Pfeiffer, R.Lienhart, G. Kühne, W. Effelsberg: The MoCA Project - Movie Content Analysis Research at the University of Mannheim, Informatik '98, pp. 329-338, 1998.

# A Geometrical Key-Frame Selection Method Exploiting Dominant Motion Estimation in Video

Brigitte Fauvet<sup>1</sup>, Patrick Bouthemy<sup>1</sup>, Patrick Gros<sup>2</sup>, and Fabien Spindler<sup>1</sup>

<sup>1</sup>IRISA/INRIA, Campus Universitaire de Beaulieu, 35042, Rennes cedex, France

<sup>2</sup>IRISA/CNRS, Campus Universitaire de Beaulieu, 35042, Rennes cedex, France

<http://www.irisa.fr/vista>

**Abstract.** We describe an original method for selecting key frames to represent the content of every shot in a video. We aim at spatially sampling in a uniform way the coverage of the scene viewed in each shot. Our method exploits the computation of the dominant image motion (assumed to be due to the camera motion) and mainly relies on geometrical properties related to the incremental contribution of a frame in the considered shot. We also present a refinement of the proposed method to obtain a more accurate representation of the scene, but at the cost of a higher computation time, by considering the iterative minimization of an appropriate energy function. We report experimental results on sports videos and documentaries which demonstrate the accuracy and the efficiency of the proposed approach.

## 1 Introduction and Related Work

In video indexing and retrieval, representing every segmented shot of the processed video by one appropriate frame, called key-frame, or by a small set of key-frames, is a common useful early processing step. When considering fast video content visualization, the selection of one frame per shot, typically the median frame, could be sufficient to avoid visual content redundancies ([1], [2]). On the other hand, key-frames can also be used in the content-based video indexing stage to extract spatial descriptors to be attached to the shot and related to intensity, color, texture or shape, which enables to process a very small set of images while analyzing the whole shot content. Considering only the median image of the shot is obviously too restrictive in that case. The same holds for video browsing. Another important issue is video matching based on feature similarity measurements. When addressing video retrieval, key frames can be used to match the videos in an efficient way. As a consequence, extracting an appropriate set of key-frames to represent a shot, is an important issue. Several approaches have been investigated to extract key frames. A first category exploits clustering techniques ([3], [4]). Different features can be considered (dominant color, color histogram, motion vectors or a combination of them). Selected images are then representative in terms of global characteristics. Another class of methods consists in considering key frame selection as an energy minimization problem ([5], [6]) that is generally computationally expensive. There are also the sequential methods [7], [12], that somehow consider frame-by-frame differences. If the cumulated dissimilarities are larger than a given threshold, a new key frame is selected. With such methods, the number of selected key frames depends on the chosen threshold value.

In this paper, we present an original key frame selection method that induces very low computation time and which is not dependent on any threshold or parameter. Contrary to usual approaches involving a temporal sampling of the shot, its principle is to get an appropriate overview of the scene depicted in the shot by extracting a small set of frames corresponding to a uniform spatial sampling of the coverage of the scene viewed by the moving camera. This method relies on geometrical criteria and exploits the computation of the camera motion (more specifically, of the dominant image motion) to select the best representative frames of the visualized scene. One of the interests of this approach is to be able to handle complex motions such as zooming in the key-frame selection process. Another important feature of our method consists in considering geometrical properties only, which provides an accurate and efficient solution. The remainder of the paper is organized as follows. In Section 2, we present the objectives of this work. Section 3 describes the proposed method called direct method. Section 4 is concerned with an iterative method to refine the previous solution based on an energy minimization. Results are reported in Section 5 and Section 6 concludes the paper.

## 2 Objectives

Our goal is to account for the complete visualized scene within the shot with the minimal number of key-frames, in order to inform on the visual content of each shot as completely as possible but in the most parsimonious way. Key frames are provided in order to enable fast visualization, efficient browsing, similarity-based retrieval, but also further processing for video indexing such as face detection or any other useful image descriptor extraction.

Camera motion when video is acquired can involve zooming, panning or traveling motion. Therefore, information supplied by the successive frames is not equivalent. For this reason, it is required to choose an appropriate distribution of the key frames along the shot which takes into account how the scene is viewed, while being able to handle complex motions.

The last objective is to design an efficient algorithm since we aim at processing long videos such as films, documentaries or TV sports programs. That is why we do not want to follow approaches involving the construction of images such as mosaic images [11], or “prototype images”, but we want to select images from the video stream only. Beyond the cost in computation time, reconstructed images would involve errors which may affect the subsequent steps of the video indexing process.

## 3 Key-Frame Selection Based on Geometric Criteria

We assume that the video has been segmented into shots. We use the shot change detection method described in [9] which can handle both cuts and progressive transitions in the same framework. The dominant image motion is represented by a 2D affine motion model which involves six parameters and the corresponding flow vector at point  $p(x,y)$  is given by:  $\omega_0=(a_1+a_2x+a_3y, a_4+a_5x+a_6y)$  varying over time. It is assumed to be due to the camera motion, and it is estimated between successive images at each time instant with the real-time robust multi-resolution method described in



[10]. The shot change detection results from the analysis of the temporal evolution of the (normalized) size of the set of points associated with the estimated dominant motion [9].

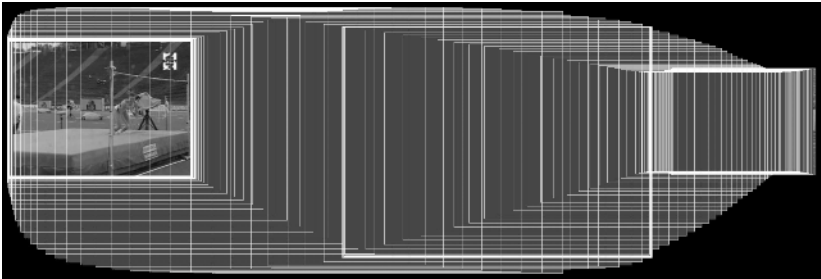
### 3.1 Image Transformation Estimation

In order to evaluate the potential contribution (in terms of scene coverage) of every new frame of a given shot, we have to project all the frames in the same coordinate system, e.g., the one corresponding to the first frame of the shot. To this end, we need to compute the transformation between the current frame of the shot and the chosen reference image frame. To do this, we exploit the dominant image motion computed between successive images in the shot change detection step, more specifically the parameters of the 2D affine motion model estimated all along the sequence. The transformation between the current frame  $I_t$  and the reference frame  $I_{ref}$  (in practice, the first frame of the shot) is obtained by first deriving the inverse affine model between  $I_t$  and  $I_{t-1}$  from the estimated one between  $I_{t-1}$  and  $I_t$ , then by composing the successive instantaneous inverse affine models from instant  $t$  to instant  $t_{ref}$ . Finally, we retain three parameters only of the resulting composed affine motion model, to form the transformation between frames  $I_t$  and  $I_{ref}$ , that is, the translation and the divergence parameters :  $\delta_1 = a_1^{t \rightarrow ref}$ ,  $\delta_2 = a_4^{t \rightarrow ref}$ ,  $\delta_3 = (a_2^{t \rightarrow ref} + a_6^{t \rightarrow ref}) / 2$ .

Aligning the successive frames with the three-parameter transformation (i.e.,  $(x', y') = (\delta_1 + (\delta_3 + 1)x, \delta_2 + (\delta_3 + 1)y)$ ) makes the evaluation of the contribution of each frame easier since the transformed frames thus remain (horizontal or vertical) rectangles, while being sufficient for that purpose.

### 3.2 Global Alignment of the Shot Images

All the successive frames of a given shot are transformed in the same reference system as explained in the previous subsection. The envelop of the cumulated transformed frames forms what we call “the geometric manifold” associated to the shot. Obviously, the shape of this manifold depends on the motion undergone by the camera during the considered shot and accounts for the part of the scene space spanned by the camera. This is illustrated by the example (Fig.1) where the camera tracks an athlete from right to left (with zoom-out and zoom-in effects) during her run-up and high-jump.

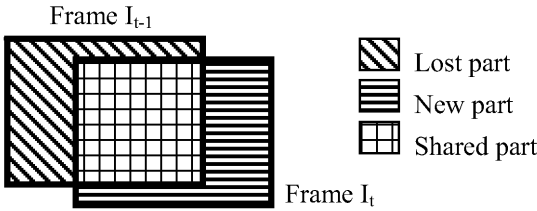


**Fig. 1.** Geometric manifold associated to a shot (the frames have been sub-sampled for clarity of the display); the last image of the shot is included.

We aim at eliminating redundancy between frames in order to get a representation of the video as compact as possible. We will exploit geometric properties to determine the number of frames to be selected and their locations in the shot.

### 3.3 Description of the Geometric Properties

We have now to evaluate in an efficient way the scene contribution likely to be carried by each frame. To define them, we consider that the frames have been first transformed in the same reference coordinate system as explained above. Then, every frame involves three kinds of scene information: a new part, a shared part and a lost part (see Fig.2).



**Fig. 2.** Definition of the three scene information parts related to frame  $I_t$ .

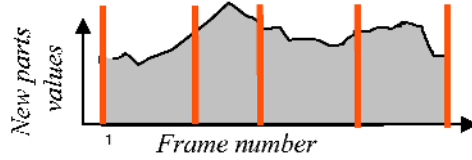
As a matter of fact, we are only interested in considering the geometric aspect of these three sets of scene information. The new part is the part of the scene brought by the current frame and which was not present in the previous frame. Conversely, the lost part is the one only supplied by the previous frame. Finally, the shared part is common to the two successive frames and corresponds to the redundant information. The surfaces of the lost part, of the shared part and of the new part will be respectively denoted by  $\sigma_L$ ,  $\sigma_S$  and  $\sigma_N$ .

These respective contributions of the two images to the description of the scene can be translated in terms of information. Let us choose the pixel as the basic information element carried by an image. The information present in the three parts  $\sigma_L$ ,  $\sigma_S$  and  $\sigma_N$  of an image are thus proportional to the number of pixels used to represent these surfaces. In the case of a zoom between the two images, the common portion will be described with more pixels in the zoomed image which thus brings more information than the other one. This is conforming to common sense.

In practice, the computation of these three information quantities requires the determination of polygons (see Fig.2) and the computation of their surface (number of pixels), which requires a low computation time.

### 3.4 Determination of the Number of Key Frames to Be Selected

The first step is to determine the appropriate number of key frames to select before finding them. We need to estimate the overall scene information, in a geometric sense, supplied by the set of frames forming the processed shot. It corresponds to the surface (denoted  $\Sigma_M$ ) of the geometric manifold associated to the shot.



**Fig. 3.** Plot of the new information parts of the successive frames of the shot. Selecting key-frames (located by vertical segments along the temporal axis) of equivalent geometric contribution amounts to get strips of equivalent size partitioning the grey area equal to the surface  $\Sigma_M$ .

A simple way to compute this surface  $\Sigma_M$  is to sum the new-parts surfaces  $\sigma_N$  of the  $N_p$  successive frames of the shot. Selected key-frames are expected to bring an equivalent geometric contribution to the scene coverage (see Fig.3). Then, the number  $N^*$  of key-frames is given by the closest integer to the ratio between the surface  $\Sigma_M$  and the size of the reference frame  $\Sigma(I_1)$  which is given by the number of pixels of the reference image  $I_1$ .

### 3.5 Key-Frame Selection

The  $N^*$  key-frames to find are determined according to the following criterion. We construct the cumulated function  $S(k)$  by successively adding the new scene information supplied by the  $N_p$  successive frames of the shot:

$$S(k) = \sum_{j=1}^k \sigma_N(j) \quad \text{with} \quad \begin{cases} \sigma_N(1) = \Sigma(I_1) \\ S(N_p) = \Sigma_M \end{cases} \quad (1)$$

The selection principle is to place a new key frame each time the function  $S(k)$  has increased of a quantity equal to the expected mean contribution given by  $\Sigma_M/N^*$ . This is equivalent to what is commented and illustrated in Fig. 3. Since we have to finally deal with entire time values, we consider in practice the two frames  $I_{k-1}$  and  $I_k$ , such that,  $k-1 \leq t_i \leq k$ , where the value  $t_i$  is the real position of the  $i^{\text{th}}$  key frame to select. The selected frame between these two frames is the one corresponding to the cumulated scene information value,  $S(k-1)$  or  $S(k)$ , closest to the appropriate multiple of the

mean contribution defined by:  $M(i) = i \times \frac{\Sigma_M}{N^*}$ . In addition, we take the first frame of the shot as the first key-frame.

## 4 Key-Frame Selection Refinement

The proposed method provides an efficient way to select appropriate key-frames in one pass as demonstrated in the results reported below. Nevertheless, one could be interested in refining the key-frame localizations, if the considered application requires it and does not involve a too strong computation time constraint. In that case, the solution supplied by the method described in Section 3, can be seen as an initial one which is then refined by an iterative energy minimization method as explained below.

#### 4.1 Energy Function

Let us consider the set of  $N^*$  key frames as a set of sites:  $X=\{x_1, x_2, \dots, x_{N^*}\}$ , with the following symmetric neighborhood for each site  $x$  (apart from the first and the last ones):  $V_x = \{x-1, x+1\}$  (1D non-oriented graph). In case this method would not be initialized with the results supplied by the direct method of Section 3,  $N^*$  would be still determined as explained in subsection 3.4. Let  $T=\{t_1, t_2, \dots, t_{N^*}\}$  be the labels to be estimated associated to these sites, that is the image instants to be selected. They take their values in the set  $\{1, \dots, N_p\}$ , where  $N_p$  is the number of frames of the processed shot (with the constraint  $t_1 = 1$ ).

Let us assume that they can be represented by a Markov model as follows. The observations are given by the scene information parts  $\sigma_N = \{\sigma_N(1), \dots, \sigma_N(N_p)\}$  and  $\sigma_s = \{\sigma_s(1), \dots, \sigma_s(N_p)\}$ . We have designed an energy function  $U(T, \sigma_s, \sigma_N)$  specifying the Markov model and composed of three terms:  $U1(T)$ ,  $U2(T, \sigma_s)$  and  $U3(T, \sigma_N)$ . The first term will express the temporal distance between the initial key-frames and the new selected ones. It aims at not moving the key frames too far from the initial ones  $\{t_{x_i}^0\}$ . The second term aims at reducing the shared parts between the key-frames while not being strictly null in order to preserve a reasonable continuity. The third term will be defined so that the sum of the new parts of the selected key-frames is close to the surface  $\Sigma_M$  of the shot manifold. The energy function is then given by:

$$U(T, \sigma_s, \sigma_N) = U1(T) + \beta U2(T, \sigma_s) + \gamma U3(T, \sigma_N) \quad \text{with :} \quad (2)$$

$$U1(T) = \sum_{x_i} |t_{x_i} - t_{x_i}^0|, \quad U2(T, \sigma_s) = \left| \sum_{k=1}^{N^*} \sigma_s(k) - \frac{N^* \Sigma(I_1)}{\alpha} \right|,$$

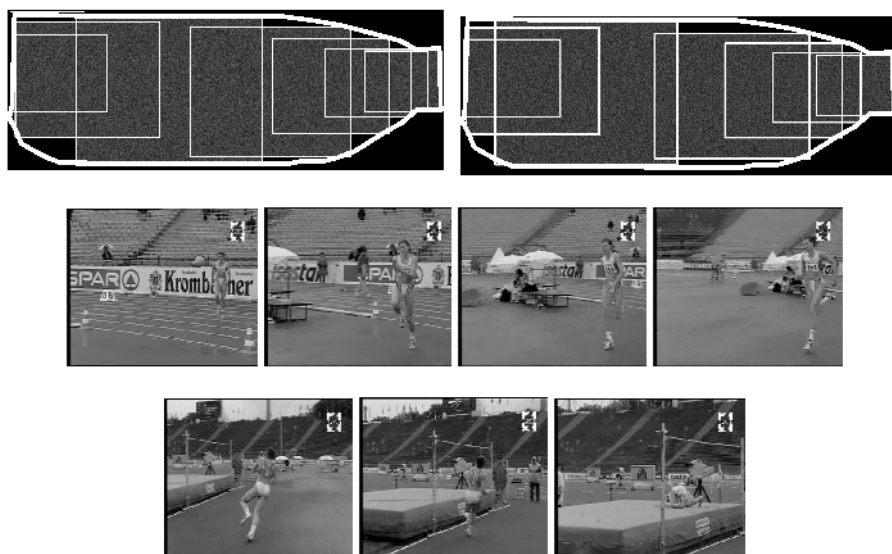
$$\text{and } U3(T, \sigma_N) = \left| \sum_{k=1}^{N^*} \sigma_N(k) - \Sigma_M \right|$$

$\beta$  and  $\gamma$  are weighting parameters (automatically set using appropriate relations) controlling the respective contributions of the three terms.  $\alpha$  is set according to the value of  $N^*$  (typically  $\alpha = 4$  in the reported experiments). Let us note that the cliques  $\langle x_i, x_{i+1} \rangle$  are involved in the computation of  $U2$  and  $U3$  through  $\sigma_s$  and  $\sigma_N$ .

We minimize the energy function  $U(T, \sigma_s, \sigma_N)$  using a simulated annealing technique in order not to be stuck in local minima. We can afford it since we deal with a very small set of sites. We use a classical geometric cooling schedule to decrease the so-called temperature parameter.

## 5 Experiments

We have processed several videos of different types. Due to page number limitation, we only report in details two representative examples: a sport sequence and a documentary one.



**Fig. 4.** Results of the direct method (top left) and iterative method (top right). The whole manifold is displayed in Fig.1; the seven key-frames obtained with the iterative method.

The first reported example is a shot of a high jump in an athletics meeting video. It involves a camera panning motion with zoom-in and zoom-out operations, to track the athlete during her run-up and high jump. The corresponding geometric manifold is shown in Fig.1.

We have first applied the direct method described in Section 3; the results are shown in Fig.4 (top left). The number of selected key frames is  $N^*=7$ . We can notice that they correctly cover the scene viewed in the shot while accounting for the zoom motion and associated changes of resolution.

We have then applied the iterative method introduced in Section 4 and obtained the results displayed in Fig.4. Selected locations of key-frames are slightly modified and redundancy is further decreased as indicated in Table1. In order to objectively evaluate the results, we use the following criteria for performance analysis:

$$Ca = \sum_{i=1}^{N^*} \sigma_N(i), \quad Cb = \sum_{i=1}^{N^*} \sigma_S(i). \quad (3)$$

The term  $Ca$ , in relation (3), represents the cumulated new information parts of the set of selected key frames. This term corresponds to the estimated coverage of the visualized scene and it must be maximized.

The second criterion  $Cb$  evaluates the cumulated intersections of the selected key-frames, their redundancies, and it must be minimized.

This comparison has been carried out on five different sequences and is reported in Table 1. We have also considered an equidistant temporal sampling of the shot with the same number of key-frames.

**Table 1.** Performance analysis by comparing results obtained with the direct method (Section 3), the iterative method (Section 4) and a temporally equidistant sampling. Values are normalized with respect to results obtained with the latter one. The content of the processed sequences involves: athletics (S1 (see Fig4) and S2), soccer (S3), interview (S4) and documentary (S5).

	Temporal Sampling (Ca, Cb)	Direct method (Ca, Cb)	Iterative Method (Ca, Cb)
S1	(100, 100)	(105.3, 92.7)	(105.5, 92.4)
S2	(100, 100)	(104.7, 94.2)	(107.0, 91.4)
S3	(100, 100)	(101.2, 98.6)	(108.2, 90.9)
S4	(100, 100)	(104.6, 99.8)	(130.5, 98.6)
S5	(100, 100)	(323.0, 63.6)	(327.4, 61.3)

The performance improvement of the proposed methods is clearly demonstrated in Table 1, especially for sequences S4 and S5. For the sequence S5, we also provide the display of the selected key-frames in Fig.5. Our approach is able to adapt the location of the key-frames to the evolution of the camera motion which mainly occurs in the middle of the shot to track the people turning at the cross-road. On the other hand, the camera is mainly static when the two people are approaching in the first part of the shot and are receding in the last part of the shot.



**Fig. 5.** Comparison of the selection of the key-frames obtained with the three methods applied to the S5 sequence. White line delimits the geometric manifold we want to cover. Top row: temporal sampling method; middle row: direct method, bottom row: iterative method. The images of the selected key-frames are displayed.  $N^*=4$ .

## 6 Conclusion

We have presented an original and efficient geometrical approach to determining the number of key-frames required to represent the scene viewed in a shot and to select them within the images of the shot. The image frames are first transformed in the same reference system (in practice, the one corresponding to the first image), using the dominant motion estimated between successive images, so that geometrical information specifying the contribution of each image to the scene coverage can be easily computed. Two methods have been developed. The direct method allows us to solve this problem in one-pass. Results can be further refined by the iterative method which amounts to the minimization of an energy function. Results on different real sequences have demonstrated the interest and the satisfactory performance of the proposed approach. We can choose the iterative method if getting better accuracy is prevailing while computation time constraint is less important.

**Acknowledgements.** This work was partly supported by the French Ministry of Industry within the RIAM FERIA project. The videos were provided by INA.

## References

1. Y.Tonomura, A. Akutsu, K. Otsuji, T. Sadakata: videoMAP and videospaceicon: tools for anatomizing video content, INTERCHI '93, ACM Press, pp 131-141.
2. B. Shahrar, D.C. Gibbon: Automatic generation of pictorial transcript of video programs, Proc. SPIE Digital Video Compression: Algorithms and Technologies, San Jose, CA, 1995, pp. 512-519.
3. Y. Zhuang, Y. Rui, T.S. Huang, S. Mehrotra: Adaptative key frame extraction using unsupervised clustering, Proc 5<sup>th</sup> IEEE Int. Conf. on Image Processing, Vol.1, 1998.
4. A. Girgensohn, J. Boreczky: Time-constrained key frame selection technique, in IEEE International Conference on Multimedia Computing and Systems, 1999.
5. H.C. Lee, S.D. Kim: Iterative key frame selection in the rate-constraint environment, Image and Communication, January 2003, Vol 18, n°1, pp.1-15.
6. T. Liu, J. Kender: Optimization algorithms for the selection of key frame sequences of variable length, 7<sup>th</sup> European Conf. on Computer Vision, Dublin, May 2002, Vol LNCS 2353, Springer Verlag, pp. 403-417.
7. M.M. Yeung, B. Liu: Efficient matching and clustering of video shots, Proc. ICIP'95, Vol.1, 1995, pp. 338-342.
8. A. Aner, J. Kender: Video summaries through mosaic-based shot and scene clustering, 7<sup>th</sup> European Conference on Computer Vision, Dublin, May 2002, Vol LNCS 2353, Springer Verlag, pp 388-402.
9. J.M. Odobez, P. Bouthemy, Robust multi-resolution estimation of parametric motion models. Journal of Visual Communication and Image Representation, 6(4):348-365, Dec. 1995.
10. P. Bouthemy, M. Gelgon, F. Ganansia. A unified approach to shot change detection and camera motion characterization. IEEE Trans. on Circuits and Systems for Video Technology, 9(7):1030-1044, October 1999.
11. M.Irani, P. Anandan: Video indexing based on mosaic representations, IEEE Trans. on Pattern Analysis and Machine Intelligence, 86(5):905-921, May 1998.
12. J. Vermaak, P. Pérez and M. Gangnet, Rapid summarization and browsing of video sequences, British Machine Vision Conf., Cardiff, Sept. 2002.

# Extraction of Salient Features for Image Retrieval Using Multi-scale Image Relevance Function

Roman M. Palenichka, Rokia Missaoui, and Marek B. Zaremba

Dept. of Computer Science and Engineering  
Université du Québec, Gatineau, Québec, Canada  
{palenich, missaoui, zaremba}@uqo.ca

**Abstract.** The goal of the feature extraction method presented in this paper was to obtain a concise, robust, and invariant description of image content for image retrieval. The solution of this problem is chosen in the form of a visual attention operator, which can measure the saliency level of image fragments and can select a set of most salient image objects (feature vectors) for concise image description. The proposed operator, called image relevance function, is a multi-scale non-linear matched filter, whose local maxima provide the most salient image locations. A feature vector containing both local shape features and intensity features is extracted and normalized at each salient maximum point of the relevance function. The testing results of this method for retrieval of synthetic and real database images are provided.

## 1 Introduction

Although existing techniques for content-based image retrieval (CBIR) are quite diversified and sophisticated, their effectiveness and the retrieval time are not satisfactory in many application areas [1, 2]. The current approaches to CBIR can be roughly divided into two categories relatively to the image content description: computation of global features and computation of local features with their relationships. The first approach has obvious limitations in image retrieval since global features such as color histograms cannot capture all image fragments having different local characteristics. On the other hand, if more global features are involved the computational complexity of the image description and retrieval time increase significantly. Moreover, the accuracy and robustness of feature extraction will decrease. The semantic (image interpretation) gap still persists in many methods for CBIR, especially when using global features of one type only [1, 2]. Another major concern with feature extraction is the invariance problem because affine geometrical transformations of images change substantially the feature values if the features are not invariant to such transformations. Therefore, local invariant features, which cover the most salient image fragments, will give adequate and sufficient description of the image content for the retrieval purposes. This is a relatively new approach to image description in CBIR, which was referred to as an approach of salient image features [1, 2].

The approach of salient features and focusing attention during image analysis has been developed independently during many past years in order to perform effective and time-efficient search for objects of interest by attention focusing [3-6]. Relatively



recently it has been proposed by several researchers to cope with the problems of feature extraction in CBIR [7-11]. The underlying idea consists in focusing attention on the most salient image fragments or objects of interest, which are stable to intensity and shape affine (geometrical) transformations and capture well the overall image content at the same time. This is a biologically inspired approach that models some basic elements of visual perception in humans and animals, especially fast visual search for objects of interest. Invariance to shape and intensity transformations such as translation, scaling, and rotation is essential for robust image retrieval using salient shape features [7, 8]. Given salient fragments a complex image object can be represented in terms of these fragments and their relationships. On the other hand, the semantic gap can be narrowed by such an intermediate content representation, which is close to natural description of images containing real world objects.

However, some basic issues in the application of visual attention mechanisms and feature extraction remain open in CBIR. One major problem is the absence of a measure of saliency of image fragments or features and, respectively, no knowledge which fragments have to be selected for adequate image description. Moreover, distances currently used to compare images are not well suited to the approach of salient features, especially, they are not matching subjective similarity measures for images [1, 2].

The feature extraction method described in this paper is an attempt to develop a CBIR framework based on the salient feature approach, which could eliminate as much as possible the aforementioned drawbacks. This is realized through the introduction of an *image relevance function* (IRF), an image operator, which can measure the saliency of image fragments and extract invariantly and in a robust way both local shape and local intensity features (color and texture are included). Moreover, the IRF approach to CBIR offers also a matching between the image content extraction and similarity measurement between images. The introduced IRF provides an image description in the form of a set of most *salient image objects* each of them being a concise description of respective salient image fragment. Each feature vector includes invariant planar shape features, geometry features such as relative position, local scale (size) and local orientation and color (intensity) features, which may include texture features.

## 2 Representation of Image Content by Salient Image Objects

The proposed IRF method provides a description of image content as a set of  $M$  most salient image objects. Each salient object as a feature vector is associated with its own salient fragment (neighborhood) centered at a particular local maximum of the IRF. The IRF is defined generically as an image operator, which is taking local maximal values at centers of salient fragments describing objects of interest and hence the image content as a set salient objects with relative positions.

In the IRF framework, invariant (to position, size, and rotation) planar shape features are extracted independently and are used along with intensity (color) and texture features. The local shape description consists of invariant planar shape features and the parameters of the respective affine transforms in order to be able to

distinguish between them. Thus a complete feature vector  $\xi^l = \{\xi_1^l, \dots, \xi_L^l\}$  of  $L$  features is associated with  $l$ th salient maximum point of the IRF. We call it salient image object because it has a particular location in image, distinct size, local shape and intensity description, and is often associated with a real-world object or its salient part. For example, three corners (as salient image objects) with their relative positions, sizes, and color can describe concisely a triangle. Therefore, this approach provides an intermediate image representation between the real-world objects and a concise image description for CBIR.

General flowchart for salient feature extraction and database querying using the IRF approach is shown in Fig. 1. The IRF is applied first to a query image and the query image content is extracted as a set of  $M$  most salient image objects,  $Q = \{\xi^1, \dots, \xi^M\}$ . Simultaneously, some parameters in the IRF are adapted to measure the location saliency in database images using processing results obtained for the current query image. The adapted IRF is applied to each image in a given database and the image content as a set  $P = \{\pi^1, \dots, \pi^N\}$  of  $N$  salient objects is determined for each database image. The two contents are compared with each other using a so-called *Distributed Matching Distance* (DMD). The distance DMD is introduced in order to establish correspondence between two sets of salient image objects extracted by the IRF approach. In image matching, other distances between two sets of points as image feature vectors are also known. The most popular are the Hausdorff distance and the Procrustes distance of the least-squares type [12, 13]. The first one is too sensitive to outliers and occlusions in images whereas the latter is computationally intensive because of using alignments with respect to geometrical transformations.

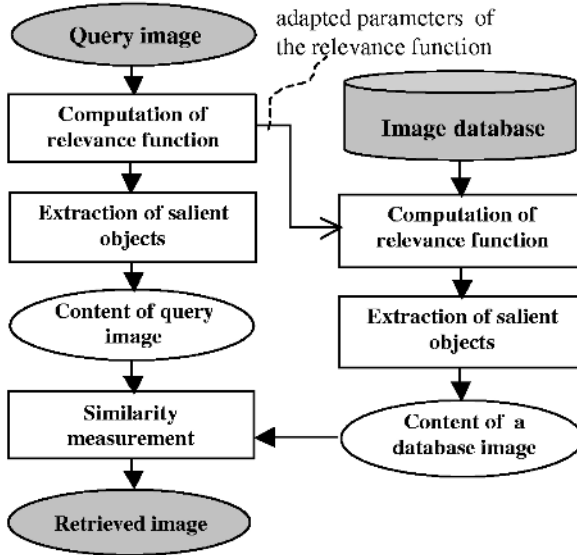


Fig. 1. Flowchart for CBIR using extraction of salient image objects by IRF.

Therefore, we propose to use the distance DMD as a two-level asymmetrical distance, which takes  $M$  most salient objects from the content of a query image,  $\mathbf{Q}$ , and considers  $N$  ( $N > M$ ) most salient objects from the content of a current database image,  $\mathbf{P}$ . The condition  $N > M$  is chosen because the set of  $M$  first and most salient objects in a database image can be different by only a few of them as compared to the first (most salient)  $M$  objects of the query image due to possible distortions and transformations of various types including local occlusions. A subset of  $M$  most similar pair of image objects is determined as best matching subset by single distances between two salient objects, each of them taken from respective two sets,  $\mathbf{Q}$  and  $\mathbf{P}$ :

$$DMD(\mathbf{Q}, \mathbf{P}) = \min_{k \in \Phi_{(N, M)}} \left\{ \sum_{m=1}^M d(\xi^m, \pi_k^m) \right\}, \quad (2.1)$$

where  $d(.,.)$  is a distance between two salient image objects and  $\Phi_{(N, M)}$  is the set indices of all possible combinations of  $N$  objects taken by  $M$  objects each from the image content  $\mathbf{P}$ . An Euclidian distance or other suitable distances in metric spaces can be used as the distance  $d(.,.)$ . In our method, the distance between two salient objects is computed as a weighted Euclidean distance,

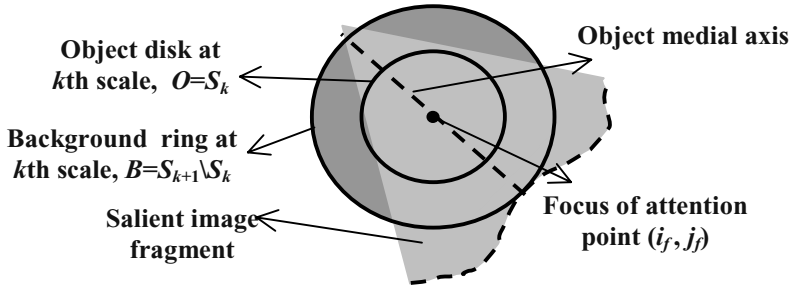
$$d^2(\xi_m, \xi_n) = \sum_{l=1}^L w_l^2 \cdot (\xi_l^m - \xi_l^n)^2, \quad (2.2)$$

where  $\{w_l\}$  are weights, which are used in order to insure a proper contribution of features to the real (desired) image similarity. The more important is a feature for the similarity establishment (with respect to a subjective criterion, i.e. a human observer) the higher will be its weight value. In this case, the distance in Eq.(2.2) can be considered as a particular case of Mahalanobis distance used in some CBIR methods [1].

### 3 Extraction of Salient Objects Using Image Relevance Function

In order to address the aforementioned problems in CBIR it is suggested to apply an improved version of a model-based IRF initially described in the context of object detection in diagnostic imaging [14]. The introduction of IRF for the salient feature extraction allows to measure explicitly the saliency level of image locations in accordance to their relevance in image description. Since the IRF approach initially was designed to process gray-scale images, the three color components of a color image have to be transformed to a gray-scale value in each image point. Better results of salient feature extraction can be achieved through a problem-oriented clustering transformation of the color components, such as the Fisher linear discriminant analysis, addressing the pattern classification problem based on color feature transformation [14].

For the purpose of multi-scale image analysis, a formal definition of scales is used: a structuring element at scale  $m$  of a *uniform scales system* is formed by the morphological dilation (denoted  $\oplus$ ) by  $S_0$ ,  $S_k = S_{k-1} \oplus S_0$ ,  $k=1, 2, \dots, K-1$ , where  $K$  is the total number of scales and the structuring element  $S_0$  defines the minimal scale



**Fig. 2.** Illustration of the relevance function computation for the single-scale case.

and object resolution. The structuring elements have the same shape such as the disk shape (see Fig. 2).

Localization of salient objects (image fragments) is based on a fast computation of the multi-scale IRF and determination of its local salient maxima. The positions of local maximum values of the multi-scale IRF coincide with location points of the salient image objects in a region of interest  $A$ :

$$(i_f, j_f)_l = \max_{(i,j) \in A} \max_k \{ \Phi[g(i,j), S_k], (i,j) \notin \Gamma_l \}, \quad (3.1)$$

where  $g(i,j)$  is the input gray-scale image,  $\Phi[g(i,j), S_k]$  is a non-linear matched filter at  $k$ th scale, and  $(i_f, j_f)_l$  are two coordinates of  $l$ th maximum. The region  $\Gamma_l \subset A$  corresponds to the masking region, which excludes from analysis determined maximum points.

Four simultaneous saliency conditions are considered in the design of  $\Phi(g(i,j), S_k)$ : 1) significant local contrast; 2) local homogeneity of object intensity; 3) specific object intensity range; 4) specific range of object sizes and shape of the scales  $\{S_k\}$ . The first condition is described by the absolute value for local object-to-background contrast, and the local homogeneity condition means that the intensity variance is relatively small in the object region. The intensity range means specific values for the object intensity in order to distinguish it from the background or other objects. Since the measures for contrast, homogeneity and intensity range involve object disk regions and background ring regions of a particular range, the IRF will take into account shape and scale constraints (third and fourth conditions) of the objects of interest.

Taken these conditions, the multi-scale IRF can be written at scale  $S_k$ :

$$\Phi[g(i,j), S_k] = c^2(i,j, S_k) - \alpha \cdot d^2(i,j, S_k) - \beta \cdot e^2(i,j, S_k), \quad (3.2)$$

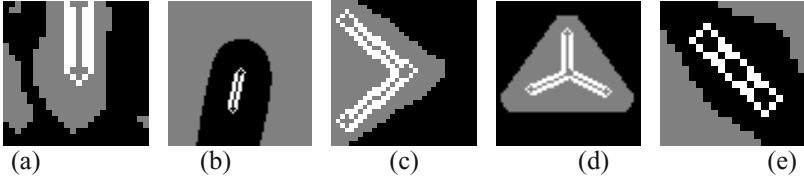
where  $c(i,j, S_k)$  is an estimate for the local contrast in point  $(i,j)$ ,  $d(i,j, S_k)$  is an estimate for intensity deviation in the object region,  $e(i,j, S_k)$  is the object intensity shift,  $\alpha$  and  $\beta$  are constraint coefficients which control the contributions of the two constraints to the overall value of IRF. The optimal values of constraint coefficients are inversely proportional to the variances of two constraints in the case of Gaussian distributions:

$\alpha = \sigma_c^2 / \sigma_d^2$  and  $\beta = \sigma_c^2 / \sigma_e^2$ . For example, the contrast estimate  $c(i, j, S_k)$  is the intensity difference,

$$c(i, j, S_k) = f_1(i, j, S_k) - f_0(i, j, Q_k), \quad (3.3)$$

where  $Q_k = S_{k+1}/S_k$ , is the background estimation region at scale  $k$ , i.e., a ring around the disk  $S_k$ .  $f_1(i, j, S_k)$  and  $f_0(i, j, Q_k)$  are the mean values of  $g(i, j)$  in regions  $S_k$  and  $Q_k$ , respectively. The standard mean square deviation was used for the estimation of  $d(i, j, S_k)$  in Eq.(3.2). The object intensity shift is measured as a deviation of the mean intensity value  $f_1(i, j, S_k)$  from the object intensity of reference.

The IRF calculation is implemented in a time-efficient manner due to the use of fast recursive filtering algorithms [14]. The computational complexity of a multi-scale linear filtering alone will be  $O(K \cdot N^2)$  operations per pixel if computed directly, where  $K$  is the scale total number,  $N \times N$  is the filter size. Recursive procedures for fast computation of linear and non-linear filters can reduce the complexity to  $O(K)$  operations per pixel, i.e., the complexity becomes independent from  $N$ .



**Fig. 3.** Examples of extracted local shape features (piecewise-linear skeletons) of salient image objects superimposed on the image fragments.

#### 4 Extraction of Salient Shape and Intensity Features

The extraction of salient features uses the intermediate IRF computation results and is computationally insignificant as compared to the IRF computation. The invariance parameters for the considered geometrical transforms (translation, scaling, and rotation) are computed with respect to the current local maximum of the IRF,  $(i_f, j_f)$ . The first parameter is absolute position of the  $l$ th salient object consisting of two coordinates  $v_l = (i_f, j_f)_l$ . The next two parameters, local scale and local orientation, which are related to a particular location  $v_l = (i_f, j_f)_l$ , are estimated using some of the intermediate results when calculating the IRF. This can be made in a fast manner as follows. The main part of local scale determination is already included in the computation of the multi-scale matched filter at  $(i_f, j_f)$ , Eq. (3.2):

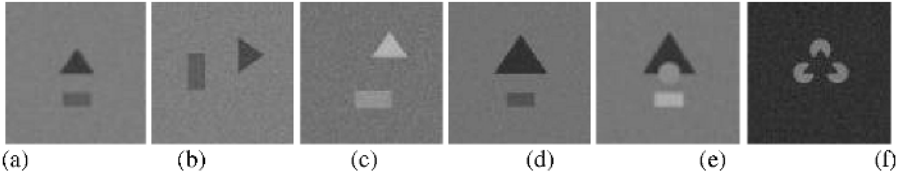
$$\rho(i_f, j_f) = \arg \max_k \left\{ c^2(i_f, j_f, S_k) - \alpha \cdot d^2(i_f, j_f, S_k) \right\}, \quad (4.1)$$

where the variables and the constant coefficient  $\alpha$  have the same meanings as in Eq.(3.2). Estimation of orientation is simple because the next maximum point  $(i_{f+1}, j_{f+1})$  in the current region of attention with the focus of attention  $(i_f, j_f)_l$  provides the orientation vector [14].

We have proposed a piecewise-linear local skeletal description of planar shapes for the salient image objects. A skeletal shape representation in the general case is a very economical approach to shape description. An object local shape is related to a particular location  $v_0$  and local scale value at that location. Given  $K$  vertices and  $K$  scale values associated with each vertex, the local planar shape (support region  $U$ ) of an object of interest located at  $v_0$  is formed by the dilation operations of skeleton straight-line segments  $\{G_{0,k}\}$  with size-variable structuring elements,  $\{S(G_{0,k})\}$ :

$$U = \bigcup_{k=1}^{K-1} G_{0,k} \oplus S(G_{0,k}) = \bigcup_{k=1}^{K-1} \bigcup_{(i,j) \in G_{0,k}} S_m^k(i, j), \quad (4.2)$$

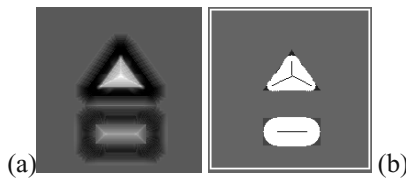
where  $S_m^k(i, j)$  is a structuring element with variable size  $m(i, j)$  as a function of point  $(i, j) \in G_{0,k}$ . The value of  $m(i, j)$  is a linear combination of the scale values at terminal vertices of  $G_{0,k}$ . In fact, Eq. (4.2) represents a method of *scale-interpolated dilation* in the piecewise-linear modeling of skeletal shapes, and  $K-1$  is the maximal topological order of the skeleton vertices.



**Fig. 4.** Examples of synthetic images used in the experiments on localization accuracy and invariant feature extraction.

The proposed IRF approach provides at the same time a method how to determine vertices for the piecewise-linear skeletal representation of object shape. Next  $K-1$  maxima of the IRF determine  $K-1$  skeleton vertices, which are connected to the vertex  $v_0$  in Eq. (4.2). Examples of detected salient image objects (fragments) in real images with superimposed piecewise-linear skeleton fragments are shown in Fig. 3.

The intensity features, which are also attached to the shape feature vector, include object mean intensity, color intensity components, local contrast, and local object variance. Some object texture features may be used as well in order to describe concisely intensity fluctuations for large scales (object sizes) [1]. The intensity features are computed by intensity averaging within the object region  $S_\rho$  and the background region  $Q_\rho = S_{\rho+1}/S_\rho$ , where  $\rho$  is the local scale determined by two maximums of IRF [14].



**Fig. 5.** Example of IRF calculation (a) and shape extraction (b) for the image in Fig. 4a.

## 5 Experimental Results

### 5.1 Accuracy and Robustness of Localization of Salient Image Objects

The first kind of experiments was the performance testing (localization accuracy and robustness against noise and occlusions) of salient object extraction using the IRF approach. The localization accuracy testing consisted of using synthesized salient image objects of known positions with added noise (random perturbations of intensity) with various contrast-to-noise ratios (see examples in Fig. 4). An example of IRF calculation and shape feature extraction is shown in Fig. 5 for the case of the original query image. Graph of the localization accuracy versus the noise level was determined experimentally (see example in Fig. 6) by adding noise with different variances to a synthetic image (Fig. 4a) with known object contrast. Analysis of these data shows good accuracy of the proposed approach for feature extraction and its robustness to noise. Similar experiments were carried out in order to test the robustness against the influence of local occlusions (see example in Fig. 4e). The occlusions influence insignificantly the accuracy unless they overlap the salient image fragments. The occlusion resistance is a remarkable property of the described IRF approach using the distance DMD due to the selection and matching of  $M$  most salient objects. The robustness against geometrical planar shape transformations was measured on the synthetic image set by a relative frequency of correct extraction of salient objects that was equal to 0.98.

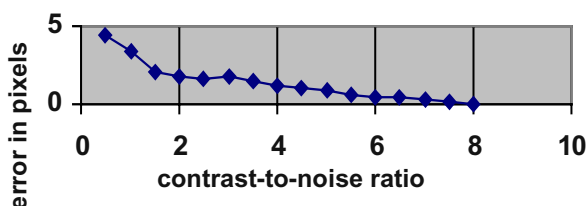
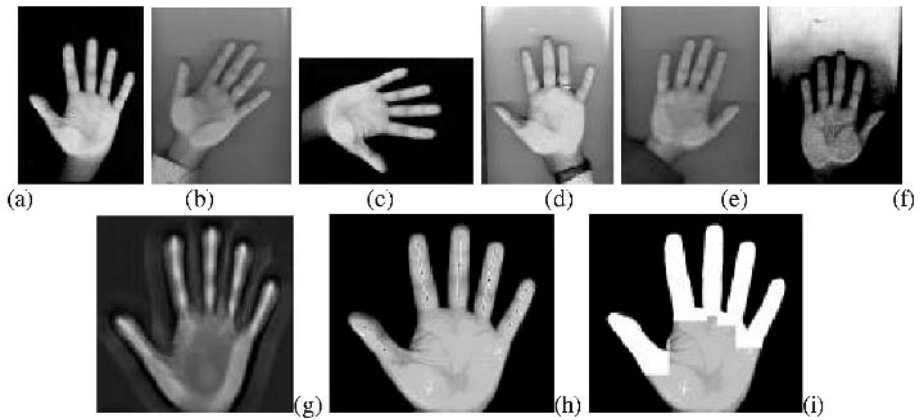


Fig. 6. Measured IRF localization accuracy for synthetic images with added noise.

On the other hand, the suggested approach can model some visual illusions, which are inherent to humans. Examples of the illusions connected to perception of planar shapes are the Kanizsa figures (see Fig. 4f). In fact, this is not a real triangle, but a collection of three disks with acute angles cut out in appropriate places. Using the IRF approach, the image with the Kanizsa triangle was retrieved as the most similar to a query image containing a triangle (Fig. 4a) when using only three most salient image objects.

### 5.2 Retrieval of Transformed Images from a Biometrical Database

The objective of this experiment was the performance evaluation of the IRF in querying databases with biometrical images. Currently, experiments were conducted with a subset taken from this database containing only hand images scanned



**Fig. 7.** Examples of hand images used in the experiments on correct retrieval rate. Results of IRF calculation and shape extraction for the image in (a) are shown in (g), (h) and (i).

**Table 1.** Correct retrieval rate (in %) in experiments using transformed query images.

Total number of involved objects	1	3	6	12	24
Scaling included	86	91	94	97	98
Scaling and rotation included	78	89	93	98	100

arbitrarily (no template marked on the surface) from different persons with total number of 64 images. Some examples are shown in Fig. 7 with marked salient object locations in Fig. 7h. The performance (correct retrieval rate) was measured as a normalized frequency of correct image retrieval using geometrical transformations of scaling and rotation with different parameters such as the scale and angle values. Each test query consisted of a selection of one different image from the database as a query image and a geometrical transformation (scaling or/and rotation) of that image before querying. The most similar image from the image database had to be selected. The correct retrieval rate (in percentage to the total number of queries with transformations) for different numbers of salient objects was estimated after selecting all the database images once at a time as a query image (Table 1).

## 6 Conclusions

The proposed method for CBIR uses a set of most salient objects for image content representation. It is based on establishing correspondence between two sets of salient objects (a query image and a database image) including their relationships inside the two images. The extraction of salient objects is performed in a fast and robust way by time-efficient calculation of the introduced IRF and the determination of its local salient maxima. The proposed concise description of image content has the following advantages in CBIR. It provides robust image retrieval in the presence of noise and



under some local distortions and occlusions. Both intensity (color) and shape features are combined to form an adequate and invariant (to geometrical transformations) image representation.

**Acknowledgments.** We are grateful to VRQ (Valorisation Recherche Québec) and Canadian Heritage for their financial support to CoRIMedia (Consortium de Recherche en Image et Multimedia).

## References

1. A. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain, „Content-based image retrieval at the end of the early years“, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 22, No. 12, pp. 1349-1380, 2000.
2. Y. Rui, T. S. Huang, and S.-F. Chang, „Image retrieval: current techniques, promising directions and open issues“, *Journal of Visual Communication and Image Representation*, Vol. 10, pp. 39-62, No. 3, 1999.
3. T. Lindeberg, „Detecting salient blob-like image structures and their scale with a scale-space primal sketch: a method for focus of attention“, *Int. Journal of Computer Vision*, Vol. 11, pp. 283-318, 1993.
4. L. Itti, C. Koch, and E. Niebur, „A model of saliency-based visual attention for rapid scene analysis“, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, pp. 1254-1259, 1998.
5. D. Reissfeld *et al.*, „Context-free attentional operators: the generalized symmetry transform“, *Int. Journal of Computer Vision*, Vol. 14, pp. 119-130, 1995.
6. H. D. Tagare, K. Toyama, and J.G. Wang, „A maximum-likelihood strategy for directing attention during visual search“, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, No. 5, pp. 490-500, 2001.
7. C. Schmid and R. Mohr, „Local gray-value invariants for image retrieval“, *IEEE Trans. Pattern Anal. and Machine Intel.*, Vol. 19, No. 5, pp. 530-535, 1997.
8. T. Tuytelaars and L. K. Van Gool, „Content-based image retrieval based on local affinity invariant regions“, *Proc. Visual'99: Information and Information Systems*, pp. 493-500, 1999.
9. N. Sebe *et al.*, „Evaluation of salient point techniques“, *Proc. Image and Video Retrieval, CIVR2002*, Vol. LNCS 2383, pp. 267-377, 2002.
10. F. Schaffalitzky and A. Zisserman, „Automated scene matching in movies“, *Proc. Image and Video Retrieval, CIVR2002*, Vol. LNCS 2383, pp. 186-197, 2002.
11. W. Wang, Y. Song, and A. Zhang, „Semantic-based image retrieval by region saliency“, *Proc. Image and Video Retrieval, CIVR2002*, Vol. LNCS 2383, pp. 29-37, 2002.
12. D. P. Huttenlocher, G.A. Klanderman, and W. Rucklidge, „Comparing images using the Hausdorff distance“, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 15 No. 9, pp. 850-863, 1993.
13. N. Duta, A. K. Jain, and M.-P. Dubuisson-Jolly, „Automatic construction of 2D shape models“, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, No. 5, pp. 433-446, 2001.
14. R. M. Palenichka, „A visual attention operator based on morphological models of images and maximum likelihood decision“, *Proc. Workshop SSPR 2002, LNCS 2396*, pp. 310-319, 2002.

# Relevance Feedback for Keyword and Visual Feature-Based Image Retrieval\*

Feng Jing<sup>1</sup>, Mingjing Li<sup>2</sup>, Hong-Jiang Zhang<sup>2</sup>, and Bo Zhang<sup>3</sup>

<sup>1</sup> State Key Lab of Intelligent Technology and Systems  
Beijing 100084, China  
jingfeng00@mails.tsinghua.edu.cn

<sup>2</sup> Microsoft Research Asia  
49 Zhichun Road, Beijing 100080, China  
{mjli, hjzhang}@microsoft.com

<sup>3</sup> State Key Lab of Intelligent Technology and Systems  
Beijing 100084, China  
dcszb@mail.tsinghua.edu.cn

**Abstract.** In this paper, a relevance feedback scheme for both keyword and visual feature-based image retrieval is proposed. For each keyword, a statistical model is trained offline based on visual features of a small set of manually labeled images and used to propagate the keyword to other unlabeled ones. Besides the offline model, another model is constructed online using the user provided positive and negative images as training set. Support vector machines (SVMs) in the binary setting are adopted as both offline and online models. To effectively combine the two models, a multi-model query refinement algorithm is introduced. Furthermore, an entropy-based active learning strategy is proposed to improve the efficiency of relevance feedback process. Experimental results on a database of 10,000 general-purpose images demonstrate the effectiveness of the proposed relevance feedback scheme.

## 1 Introduction

Image retrieval based on keyword annotations [11] could be traced back to late 1970s, mainly developed by the database management and information retrieval community. Semantics of images can be accurately represented by keywords, as long as keyword annotations are accurate and complete. The challenge is that when the size of image database is large, manual annotation of all the images becomes a tedious and expensive process. Although it is possible to use surrounding text of images in the web page to extract keyword features of the images [10], such automatically extracted keywords are far from being accurate. These facts limit the scale up of keyword-based image retrieval approaches.

On the other hand, content-based image retrieval (CBIR) [3][5][12][15] has been introduced and developed since early 1990s to support image search based on visual

---

\* This work was performed at Microsoft Research Asia. Feng Jing and Bo Zhang are supported in part by NSF Grant CDA 96-24396.

features, such as color, texture and shape. Although these features could be extracted from images automatically, they are not accurate enough to represent the semantics of images. After over a decade of intensified research, the retrieval result is still not satisfactory. The gap between visual features and semantic concepts is acknowledged to be the major bottleneck of CBIR approaches. To bridge the gap, one effective approach is to use relevance feedback.

Relevance feedback is an online learning technique used to improve the effectiveness of information retrieval systems [9]. Since its introduction into image retrieval in middle 1990's, it has been shown to provide dramatic performance improvement [3][7][12][15]. There are two key issues in relevance feedback: the choice of the learning strategy and the selection of images for the users to label. For the former issue, one of the most effective learning techniques used in relevance feedback is support vector machine (SVM) [4], which has not only strong theoretical foundations but also excellent empirical successes. An SVM classifier is trained based on the positive and negative images marked by a user and used to classify other unlabelled images into relevant and irrelevant classes [12]. For the later issue, instead of randomly selecting images, several active learning algorithms are proposed to select those most informative ones [3][12][14].

To utilize the strengths of both keyword-based and visual feature-based representations in image retrieval, a number of approaches have been proposed to integrate keyword and visual features [1][7][15]. The key issue of such integrated approaches is how to combine the two features such that they complement to each other in retrieval and/or relevance feedback processes. For example, the framework proposed in [7] uses a semantic network and relevance feedback based on visual features to enhance keyword-based retrieval and update the association of keywords with images. Zhang [15] and Chang [1] further improved this framework by updating unmarked images in addition to the marked ones using the probabilistic outputs of a Gaussian model and SVM, respectively, to perform annotation propagation.

In this paper, we propose a scheme to seamlessly integrate keyword and visual feature representations in relevance feedback. As the basis of the scheme, an ensemble of keyword models, i.e. an ensemble of SVM classifiers, is trained offline based on a small set of manually labeled images and used to propagate keywords to unlabeled ones. Comparing with the aforementioned methods [1][7][15], it has the following characteristics:

- Not only the online constructed model, i.e. an SVM classifier that separates positive images from negative ones, but also the keyword model trained offline are considered. A multi-model query refinement (MQR) technique is proposed for the combination of the two models.
- To perform relevance feedback more efficiently, an entropy-based active learning algorithm is proposed to actively select the next requests.

The organization of the paper is as follows: In Section 2, we describe the keyword propagation process based on statistical keyword models. The multi-model query refinement algorithm is introduced in Section 3. In Section 4, the active learning issue is discussed and a new entropy-based active learning algorithm is proposed. In

Section 5, we provide experimental results that evaluate all aspects of the relevance feedback scheme. Finally, we conclude in Section 6.

## 2 Keyword Propagation

A critical basis of the proposed relevance feedback scheme is the keyword models built from visual features of a set of annotated images. The models serve as a bridge that connects the semantic keyword space with the visual feature space. Similar to [1], we use SVM binary classifiers as the models, due to their sound theoretical foundations and proven empirical successes [4]. For each keyword, an SVM is trained using the images labeled with it as positive examples and other images in the training set as negative examples.

In the basic form, SVM tries to find a hyperplane that separates the positive and negative training data with maximal margin. More specifically, finding the optimal hyperplane is translated into the following optimization problem:

$$\text{Minimize: } \frac{1}{2} \|\tilde{w}\|^2 + C \cdot \sum \xi_i \quad (1)$$

$$\text{subject to: } \forall k : y_k (\tilde{w} \cdot \tilde{x}_k + b) \geq 1 - \xi_k \quad (2)$$

where  $\tilde{x}_i$  is the visual feature vector of image  $I_i$ ,  $y_i$  is equal to 1 if image  $I_i$  is labeled with the current keyword, while it is -1 otherwise.

We solve this optimization problem in its dual formulation using SVM Light [6]. It efficiently handles problems with many thousands of support vectors, converges fast, and has minimal memory requirements. Moreover, it could efficiently estimate the parameters using leave-one-out (LOO) scheme.

The key purpose of building the SVM models is to obtain the association (confidence) factor or weight of each keyword to each image in the keyword propagation process based on their visual features. As a result of the propagation, each image is associated or labeled with a set of keywords, each with a weighting factor. These weighted keywords thus form a keyword feature vector, the dimension of which is the number of total keywords in the database. This is similar to the content-based soft annotation approach proposed in [1]. To perform such soft labeling, calibrated probabilistic outputs of SVMs are required. Since standard SVMs do not provide such output, we use the method proposed by Platt [8] to resolve this issue. It trains the parameters of an additional sigmoid function to map the SVM outputs into probabilities, which are used as the confidence factor for each keyword labeling. Instead of estimating the class-conditional densities, it utilizes a parametric model to fit the posterior directly. Three-fold cross-validation is used to form an unbiased training set.

By properly incorporating spatial information into color histogram, auto-correlogram has been proven to be one of the most effective features in CBIR [5] and therefore is used as the visual features of each image in our implementation. As in [5], the RGB color space with quantization into 64 color bins is considered and the

distance set  $D = \{1, 3, 5, 7\}$  is used for feature extraction. The resulting feature is a 256-dimensional vector.

As suggested by [2], the Laplacian kernel is chosen as the kernel of SVM, which is more appropriate for histogram-based features like the one we use. Assuming  $x = (x_1, x_2, \dots, x_n)$  and  $y = (y_1, y_2, \dots, y_n)$ , the form of Laplacian kernel is:

$$k_{Laplacian}(x, y) = \exp\left(-\left(\sum_{i=1}^n |x_i - y_i|\right) / 2\sigma^2\right) \quad (3)$$

The value of  $\sigma$  is determined using cross-validation strategy. Ideally,  $\sigma$  should be tuned for each keyword. Our experimental study showed that the models were not very sensitive to the value of  $\sigma$ . For simplicity,  $\sigma$  is set to be the same for all keywords.

### 3 Multi-model Query Refinement

After the keyword propagation process, each image is represented by two types of features: keyword feature and visual feature. Denote the keyword feature vector and visual feature vector of image  $I_i$  by:  $\overrightarrow{F_i^K} = (f_{i,1}^K, f_{i,2}^K, \dots, f_{i,M}^K)$  and  $\overrightarrow{F_i^V} = (f_{i,1}^V, f_{i,2}^V, \dots, f_{i,D}^V)$  respectively.  $f_{i,j}^K$  is the probability of keyword  $K_j$  estimated using the model of  $K_j$  and  $D$  is the dimension of visual feature space which equals 256 in current implementation. The similarity score of image  $I_i$  in respect to a query keyword  $K_q$  is determined by:

$$S_i = f_{i,q}^K, 1 \leq i \leq N \quad (4)$$

The initial retrieval result is given by sorting the images in the decreasing order of their similarity scores. If the result is not satisfactory, a relevance feedback process is invoked. When the user marks a few images as feedback examples, an SVM is trained online using the visual feature of the marked images as the training set to extend the search space. More specifically, to rank an image  $I_i$  in a renewed search, the similarity of the image to the query keyword in the visual feature space is defined by  $P(K_q | I_i)$ , i.e., the probability of image  $I_i$  to be labeled with keyword  $K_q$ . A straightforward way to estimate  $P(K_q | I_i)$  is to combine the training set of online learning with that of offline learning for  $K_q$  and re-train an SVM based on the combined training set. However, considering the required real-time nature of relevance feedback interactions, the re-training process with a larger combined training set is not desirable. Instead, we compute the new model using a model ensemble scheme. That is, we have two estimations of  $P(K_q | I_i)$ : One from the model of  $K_q$  trained offline, denoted as  $P_q(I_i)$  that is equal to  $f_{i,q}^K$ ; and the other from the SVM trained online, denoted as  $P_{on}(I_i)$ . For the latter, the type and parameters of the SVM kernel are the same as those in Section 2. Considering that the number of marked images is usually small in user feedback sessions, leave-one-out strategy is used to obtain the

training set for the sigmoid fitting process. More specially, an ensemble of the two models is used to predict a more accurate estimation:

$$P(K_q | I_i) = \lambda P_{on}(I_i) + (1 - \lambda) P_q(I_i) \quad (5)$$

This model ensemble is used as the similarity of images in the renewed retrieval. That is, the refined retrieval results are obtained by re-sorting images in the decreasing order of  $P(K_q | I_i)$ .  $\lambda$  in (5) is a tunable parameter that reflects our confidence on the two estimations.  $\lambda$  is currently set to be 0.3 based on the experiments that will be introduced in Section 5. It means that  $P_q(I_i)$  is assumed to be more reliable than  $P_{on}(I_i)$ . On the other hand, the larger the value of  $\lambda$ , the more dynamic the feedback will be, though it does not necessarily lead to a faster convergence of satisfactory retrieval result.

## 4 Active Learning

As stated in Section 1, how to select more informative images from a ranked list based purely on similarities to present to a user is a crucial issue in relevance feedback to ensure efficient learning with the usually small set of training samples. The pursuing of the “optimal” selection strategy by the machine itself was referred to as active learning. In contrast to the passive learning in which the learner works as a recipient of a random data set, active learning enables the learner to use its own ability to collect training data.

Tong and Chang proposed an active learning algorithm for SVM-based relevance feedback [12]. In their algorithm, the images are selected so as to maximally reduce the size of the version space. Following the principle of maximal disagreement, the best strategy is to halve the version space each time. By taking advantage of the duality between the feature space and the parameter space, they showed that the points near the decision boundary can approximately achieve this goal. Therefore, the points near the boundary are used to approximate the most-informative points [12]. We refer this selection strategy as the nearest boundary (NB) strategy. Considering that we deal with two different SVMs (offline and online) at the same time, the NB strategy is inappropriate here. Another straightforward and widely used strategy is the most positive (MP) strategy. When MP strategy is used, the images with largest probabilities are shown to users both as current result and candidates to label. Extensive comparisons have been made between MP and NB strategy on the application of drug discovery [13]. The results show that the NB strategy is better at “exploration” (i.e., giving better generalization on the entire data set) while the MP strategy is better at “exploitation” (i.e., high number of total hits) [13]. For image retrieval, exploitation which corresponds to precision is usually more crucial than exploration.

Besides the aforementioned two strategies, we proposed a new strategy based on the information theory. Since the probability of image  $I_i$  being labeled with keyword  $K_q$  is  $P(K_q | I_i)$ , the probability of  $I_i$  being unlabeled with  $K_q$  is  $P(\overline{K_q} | I_i) = 1 - P(K_q | I_i)$ . From the information theory perspective, the entropy of this distri-

bution is precisely the information value of image  $I_i$ . Therefore, the images with maximal entropy should be selected. More specific, the entropy of  $I_i$  is:

$$E(I_i) = -P(K_q | I_i) \log P(K_q | I_i) - P(\overline{K_q} | I_i) \log P(\overline{K_q} | I_i) \quad (6)$$

where  $E(I_i)$  is maximized when  $P(K_q | I_i) = 0.5$  and the smaller the difference between  $P(K_q | I_i)$  and 0.5 the larger the value of  $E(I_i)$ . Instead of calculating entropy explicitly, we use a simpler criterion to characterize the information value of  $I_i$ . The information value (IV) of  $I_i$  is defined to be:

$$IV(I_i) = 0.5 - |P(K_q | I_i) - 0.5| \quad (7)$$

We use this maximal entropy (ME) strategy to select those images with the largest information values to ensure faster convergence to a satisfactory retrieval result in relevance feedback process.

## 5 Experimental Results

We have evaluated the proposed framework with a general-purpose image database of 10,000 images from COREL. In our experiments, ten percent of all images in the database were labeled and used to train the keyword models. The rest of images were used as ground truth as they are all categorized as well. Currently, an image is labeled with only one keyword, the name of the category that contains it. In other words, there are totally 79 keywords representing all images in the database. All these 79 keywords constitute the query set.

First, the initial retrieval was evaluated. A retrieved image is considered a match if it belongs to the category whose name is the query keyword. Precision is used as the basic evaluation measure. When the top  $N$  images are considered and there are  $R$  relevant images, the precision within top  $N$  images is defined to be  $P(N) = R / N$ .  $N$  is also called scope in the following. The average precision vs. scope graph is shown in Figure 1. Vis and Key denote visual and keyword feature-based retrieval respectively.

Moreover, (P) and (L) denote that the propagated keyword features  $\overrightarrow{F_i^K}$ s and initially labeled keyword features  $\overrightarrow{F_i^L}$ s were used. For the latter,  $\overrightarrow{F_i^L} = (f_{i,1}^L, f_{i,2}^L, \dots, f_{i,M}^L)$  is a Boolean vector, that is,  $f_{i,j}^L = 1$  if image  $I_i$  is labeled with keyword  $K_j$ , otherwise  $f_{i,j}^L = 0$ . For  $\overrightarrow{F_i^L}$ s, Hamming distance is used as the distance function. It is observed from Figure 1 that the retrieval accuracy of using the initially labeled keyword features, i.e.  $\overrightarrow{F_i^L}$ s is a little better than that of using the visual features. This means that if only the labeling information is used without propagation, the improvement of performance is marginal. At the mean time, the retrieval accuracy of using the keyword features learned from the models of keywords, i.e.  $\overrightarrow{F_i^K}$ s, is remarkably better than that of using the initial ones, which suggests the effectiveness of the SVM-based keyword propagation.

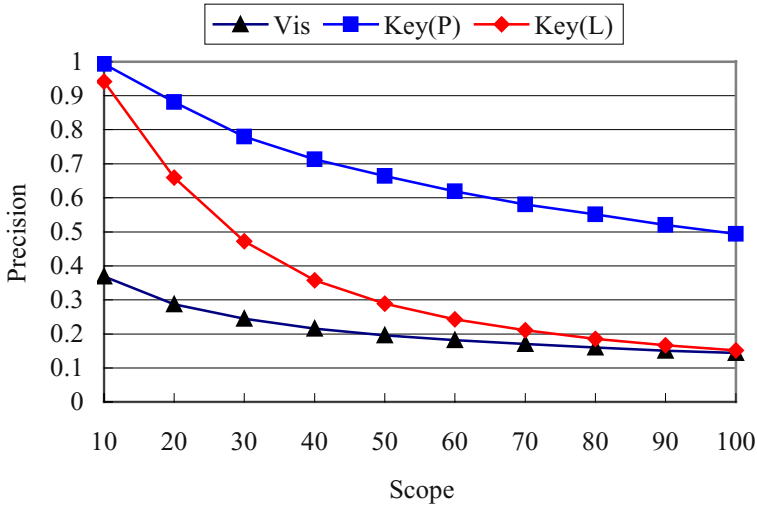


Fig. 1. Initial retrieval result comparison

Then, the proposed multi-model query refinement (MQR) algorithm was evaluated. Users' feedback processes were simulated as follows. For a query image, 5 iterations of user-and-system interaction were carried out. At each iteration, the system examined the top 10 images that have the largest informative values (IVs). Images from the same (different) category as the query image were used as new positive (negative) examples, as all images were categorized. To determine the value of  $\lambda$ , the performances of MQR with different  $\lambda$ s were compared. Generally speaking, the larger the value of  $\lambda$ , the more dynamic the feedback will be, though it does not necessarily lead to a faster convergence of satisfactory retrieval result. More specially, the accuracy vs. value of  $\lambda$  graph is used for the comparison. The accuracy is defined to be the average precision within top 50 images, i.e. average  $P(50)$ . The accuracies after 1st, 3rd and 5th rounds of feedback iterations are shown in Figure 2. Currently,  $\lambda$  is set to be 0.3 which corresponds to the peak point of the curves. Furthermore, we compared MQR with a uni-model query refinement (UQR) method that only uses the online model. Actually, UQR corresponds to  $\lambda=1$ . As we can see from Figure 2, the performance of MQR is far better than that of UQR. For example, the accuracy of MQR after three iterations is higher than that of UQR by 25%.

Finally, to show the effectiveness of active learning, three selection strategies were compared: a random selection strategy (RD), the most positive strategy (MP) and the maximal entropy strategy (ME). The accuracy vs. iteration graph is used for the comparison. Note that for RD and ME strategies, the sorting of images for evaluation and labeling is different. For evaluation, all the positive (negative) images labeled up to now are placed in top (bottom) ranks directly, while image similarity ranking are sorted by their probabilities, i.e.  $P(K_q | I_i)$  ( $1 \leq i \leq N$ ). For labeling, if the ME (or RD) strategy is used, 10 images with the highest IVs (or random selected from the database) except those labeled images were presented as retrieval result. The com-



parison results are shown in Figure 3, from which we can see that the two active learning strategies, i.e. the ME and MP strategy are consistently better than the passive learning strategy, i.e. the RD strategy after the second iteration. After five iterations the accuracy of ME (MP) is higher than that of RD by 14% (12%). In addition, the proposed ME strategy is better than the MP strategy.

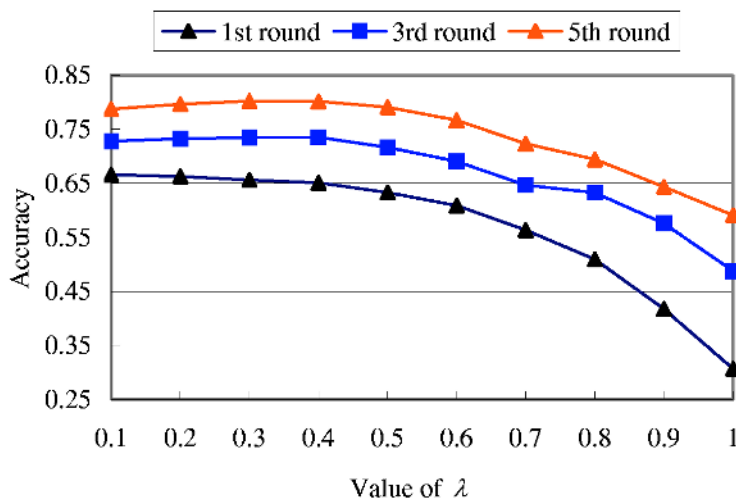


Fig. 2. The effect of different  $\lambda$  on the MQR algorithm.

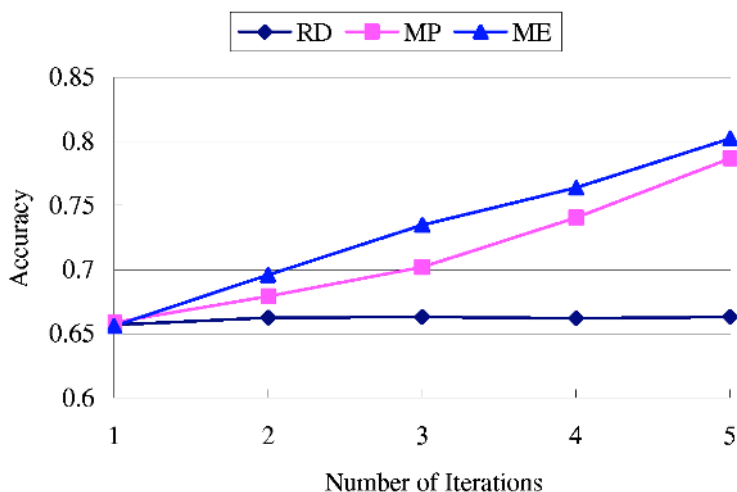


Fig. 3. Accuracy comparison of different selection strategies: RD, MP and ME denote random selecting, most positive and maximal entropy strategy respectively.

## 6 Conclusion

We have presented an effective and efficient scheme to support relevance feedback in both keyword-based and visual feature-based image retrieval. To be effective, two models are taken into account simultaneously. One is a keyword model constructed offline using SVM as the classifier and a small set of labeled images as the training set. The other is a model trained online, which is an SVM classifier that separates positive images from negative ones. A multi-model query refinement algorithm is introduced to combine the two models. To be efficient, an entropy-based active learning strategy is proposed to actively select next request images. Experimental results on a large scale database show the effectiveness and efficiency of the proposed scheme.

## References

1. Chang, E., et al, "CBSA: Content-based Soft Annotation for Multimodal Image Retrieval Using Bayes Point Machines", *IEEE Transactions on Circuits and Systems for Video Technology*, Volume 13, Number 1, January 2003, pp. 26-38.
2. Chapelle, O., Haffner, P., and Vapnik, V., "SVMs for Histogram-based Image Classification". *IEEE Transaction on Neural Networks*, 10(5), Sep. 1999, pp. 1055-1065.
3. Cox, I.J. et al, "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation and Psychophysical Experiments", *IEEE Transactions on Image Processing* 9(1), 2000, pp. 20-37.
4. Cristianini, N., Shawe-Taylor, J., "An Introduction to Support Vector Machines." Cambridge University Press, Cambridge, UK, 2000.
5. Huang, J., et al. "Image Indexing Using Color Correlogram", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, June 1997, pp. 762-768.
6. Joachims, T., "Making large-Scale SVM Learning Practical", in *Advances in Kernel Methods - Support Vector Learning*, B. Schölkopf et al (ed.), MIT-Press, 1999. pp. 169-184.
7. Lu, Y., et al, "A Unified Framework for Semantics and Feature Based Relevance Feedback in Image Retrieval Systems", *Proc. ACM International Multimedia Conference*, 2000. pp. 31-38.
8. Platt, J., "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods", in *Advances in Large Margin Classifiers*, MIT Press, 2000. pp. 61-74.
9. Salton, G., "Automatic text processing", Addison-Wesley, 1989.
10. Shen, H.T., et al, "Giving Meanings to WWW Images". *Proc. ACM International Multimedia Conference*, 2000. pp. 39-48.
11. Tamura, H. and Yokoya, N. "Image Database Systems: A Survey", *Pattern Recognition*, Vol. 17, No. 1, 1984. pp. 29-43.
12. Tong, S. and Chang, E. "Support vector machine active learning for image retrieval," *Proc. ACM International Multimedia Conference*, 2001. pp. 107-118.
13. Warmuth, M.K., Ratsch, G., Mathieson, M. and Liao, J. and Lemmen, C., "Active learning in the drug discovery process". In T.G. Dietterich, S. Becker, and Z. Ghahramani, Ed., *Adv. in Neural Inf. Proc. Sys.* 14, Cambridge, MA, 2002. MIT Press. pp. 1449-1456.

14. Zhang, C. and Chen, T., "Indexing and Retrieval of 3D models Aided by Active Learning", Demo on ACM Multimedia 2001, pp. 615-616.
15. Zhang, H. J. and Su, Z. "Improving CBIR by Semantic Propagation and Cross Modality Query Expansion", NSF Workshop on Multimedia Content-Based Information Retrieval, Paris, Sept.24-25, 2001.

# Relevance Feedback Reinforced with Semantics Accumulation

Sangwook Oh<sup>1</sup>, Min Gyo Chung<sup>2</sup>, and Sanghoon Sull<sup>1\*</sup>

<sup>1</sup> Dept. of Electronics and Computer Engineering, Korea University, Seoul, Korea  
`{osu,sull}@mpeg.korea.ac.kr`

<sup>2</sup> Dept. of Computer Science, Seoul Women's University, Seoul, Korea  
`mchung@swu.ac.kr`

**Abstract.** Relevance feedback (RF) is a mechanism introduced earlier to exploit a user's perceptual feedback in image retrieval. It refines a query by using the relevance information from the user to improve subsequent retrieval. However, the user's feedback information is generally lost after a search session terminates. In this paper, we propose an enhanced version of RF, which is designed to accumulate human perceptual responses over time through relevance feedback and to dynamically combine the accumulated high-level relevance information with low-level features to further improve the retrieval effectiveness. Experimental results are presented to demonstrate the potential of the proposed method.

## 1 Introduction

An image retrieval system solely based on low-level image features is limited in its applicability due to some reasons: for example, it is very hard to represent high-level human perceptions precisely by using low-level visual features, and those low-level features are also highly sensitive to a small change in image shape, size, orientation and color. Many active research efforts thus have been made to overcome such limitations. Among them, relevance feedback (RF) is one notable approach to integrate high-level human concepts and low-level features into image retrieval [1,2,3]. In RF approach, a user is able to interactively specify the amount of relevance between a query and resulting images, and such relevance information is used to refine the query continuously to the user's satisfaction.

Though RF is an intriguing concept for interactive image retrieval, it has one serious drawback, which is that RF ignores valuable feedback information generated from user interactions during search sessions. However, we discover that the feedback information thrown away in this way contains the important information that captures the semantics of images, thus can be more intuitive and informative description of visual content than low-level image features. Motivated by this discovery, we propose a novel RF mechanism strengthened with a capability to store and reuse the relevance feedback information effectively.

---

\* Corresponding author

Specifically, the proposed method constructs a semantic space for a large collection of images by accumulating human perceptual responses over time through relevance feedback, and dynamically combines the accumulated high-level relevance information with low-level features to further improve the retrieval effectiveness. Experimental results show that the retrieval performance of the proposed method is greatly enhanced compared with traditional RF methods and gets better and better as time passes.

The rest of this paper is organized as follows. In Sec. 2, we describe the details of the proposed method: construction of a semantic vector space for an image database, and dynamic integration of semantic information and low-level image features into RF framework. Experimental results are presented in Sec. 3 to validate some good characteristics of the proposed method. Finally, concluding remarks are given in Sec. 4.

## 2 New Image Retrieval

### 2.1 Semantic Space

Relevance feedback responses, which are generated during search sessions but destroyed immediately in traditional RF mechanisms, are now accumulated to build a semantically meaningful high-level feature space, called *semantic space* hereafter. A semantic space for an image database is represented by an  $n \times n$  matrix  $M = (m_{ij})$ , where  $n$  is the number of images in the image database and  $m_{ij}$  denotes a total of relevance scores accumulated over a certain period of time between a query image  $i$  and an image  $j$  on the image database. The more conceptually similar the two images  $i$  and  $j$ , the greater the value of  $m_{ij}$ .

Figure 1 shows a simple example of the semantic matrix  $M$  for a collection of 5 images.  $m_{ij}$  is initially zero for all images  $i$  and  $j$ , but will be filled soon with relevance values as search processes go on. For the given query image  $i$ , if the image  $j$  is marked by a user as *relevant* in a search session,  $m_{ij}$  gets updated by a particular relevance score. In Fig. 1, for example, if the image 4 is the query image and the image 5 is selected as a relevant image, then  $m_{4,5}$  is changed from 5 to  $5 + \alpha$ , where  $\alpha$  is a relevance score. Although there are many possible ways to determine relevance scores, we simply use the following scoring rule: if marked as relevant, then  $\alpha = 1$ ; otherwise, then  $\alpha = 0$ .

There are two possible ways to update (and maintain) a semantic matrix: asymmetric or symmetric. In the asymmetric update scheme, if a user establishes a relevance between a query image  $i$  and an image  $j$ , then only  $m_{ij}$  in the semantic matrix is updated to a new value, but  $m_{ji}$  remains same. On the other hand, the symmetric update scheme makes both  $m_{ij}$  and  $m_{ji}$  get updated at the same time. Although it requires further studies to investigate the properties and effectiveness of the two update schemes, the asymmetric update scheme has a tendency to return different retrieval results depending on which of the images  $i$  or  $j$  is chosen as the query image, but the symmetric update scheme tends to yield similar retrieval results irrespective of the choice of the query image. We prefer to use asymmetric update scheme because it is more general than

	Im 1	Im 2	Im 3	Im 4	Im 5
Im 1	0	1	2	1	1
Im 2	3	0	1	0	2
Im 3	3	1	0	2	4
Im 4	1	2	2	0	5
Im 5	0	1	2	3	0

**Fig. 1.** A simple example of a semantic matrix for a collection of 5 images.

the symmetric scheme and has a capability to differentiate images with subtle differences in human perception.

## 2.2 Integration of Semantic and Low-Level Features

This section will give a detailed description of how to combine high-level semantic features and low-level visual features within RF framework. Without loss of generality, we assume that an image is associated with two low-level features, color and texture, and one high-level feature, semantic feature presented in the previous section. We take color correlogram [4] as the color feature, and use Shim and Choi's method [5] to obtain the texture feature. Assume further the following symbols and definitions for convenience of explanation:

- $S_C(i, j)$ ,  $S_T(i, j)$ , and  $S_S(i, j)$  are a similarity measure of color, texture and semantic features, respectively, between two images  $i$  and  $j$ .
- $W_C$ ,  $W_T$ , and  $W_S$  are a weight associated with color, texture and semantic features, respectively.

**Similarity Computation.** The overall similarity between a query image  $i$  and an arbitrary image  $j$ ,  $S(i, j)$ , can then be calculated using the above definitions as follows:

$$S(i, j) = W_C \frac{S_C(i, j)}{\max_j S_C(i, j)} + W_T \frac{S_T(i, j)}{\max_j S_T(i, j)} + W_S \frac{S_S(i, j)}{\max_j S_S(i, j)},$$

where we use  $m_{ij}$  in the semantic matrix  $M$  for the value of  $S_S(i, j)$ . In the above equation,  $\max_j S_C(i, j)$ ,  $\max_j S_T(i, j)$ , and  $\max_j S_S(i, j)$  indicate a maximum similarity value for the corresponding image feature, and are used to normalize each similarity measure. In other words, the overall similarity  $S(i, j)$  is represented as a linear combination of individual normalized similarity measures.

**Weight Update.** For the query image presented by a user, an RF based retrieval system executes search algorithms and returns its results. The user then

views the retrieved images and judges which images are relevant and which images are not. The relevance information obtained in this way is used to dynamically update the weights of the image features as well as the semantic matrix. The weights should be changed in proportion to the relative importance of the image features.

Let  $R$  be the set consisting of  $k$  most similar images according to the overall similarity value  $S(i, j)$ , where  $k$  is the number of images the user wants to retrieve. Similarly, we define three more sets  $R_C$ ,  $R_T$ , and  $R_S$ . That is,  $R_f$  is the set of  $k$  most similar images according to the similarity measure  $S_f(i, j)$ , where  $f$  can be any of three image features,  $C$ ,  $T$ , and  $S$ . Then, the new weights for each image feature,  $f$ , are calculated using the following procedure, which is similar to the one in [1]:

1.  $W_{sum} = 0$ .
2. For each  $f$  in  $[C, T, S]$ , execute three steps below.
  - a) Initialize  $W_f = 0$ .
  - b) For each image  $p$  in  $R_f$ ,  $W_f = W_f + \alpha$  if  $p$  is in  $R$ .
  - c)  $W_{sum} = W_{sum} + W_f$ .
3. The weights obtained in Step 2 are now normalized by the total weight  $W_{sum}$  as follows:  $W_f = \frac{W_f}{W_{sum}}$ .

The above procedure implies that the more overlap between  $R_f$  and  $R$ , the larger the weight of  $W_f$ . The weights updated in the current retrieval iteration are subsequently used to return more perceptually relevant images in the next iteration.

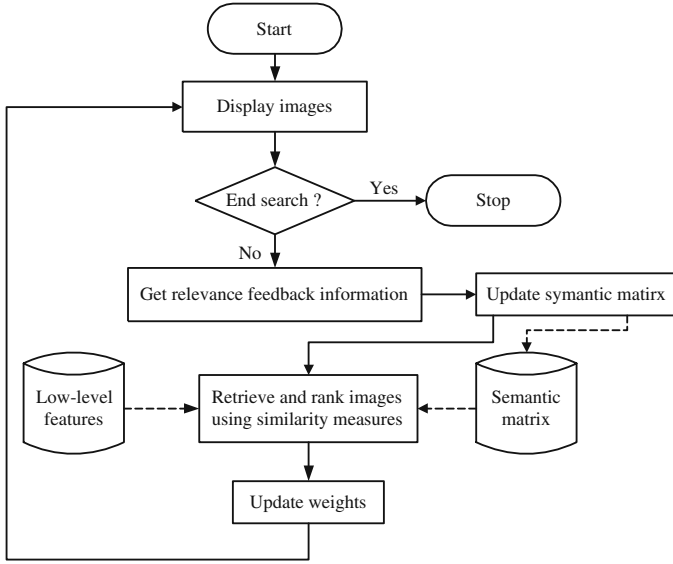
As the retrieval process continues, the weights change dynamically according to the user's information need and intention. Since the user selects only perceptually relevant images to the query through relevance feedback, as the retrieval process continues, more and more images governed by the similarity measure  $S_S(i, j)$  tend to appear on the retrieval result list. As a result, the weight for the color or texture feature gets smaller, but the weight for the semantic feature gets bigger, which means the semantic feature plays a critical role in finding relevant images quickly. Due to this favorable phenomenon, false positive rates are also dramatically reduced.

### 3 Experiments

#### 3.1 Experimental Setup

To study the effectiveness of the proposed method, we have implemented our image retrieval system that works as illustrated in Fig. 2. The initial weights are all  $\frac{1}{3}$  for color, texture and semantic features. When it comes to updating the system-wide semantic matrix, the asymmetric update scheme is chosen because it is more general than the symmetric scheme and can afford to differentiate images with subtle differences in human perception.

Our image database contains 2700 natural images, which implies the dimension of the semantic matrix  $M = (m_{ij})$  is  $2700 \times 2700$ . According to their content,



**Fig. 2.** Flowchart of the proposed retrieval system

the images in the image database are categorized into several groups: humans (celebrities, entertainers), animals, vehicles (cars, airplanes, motorcycles), stars, plants, natural scenes (sunrise, sunset, clouds, lightning), sports, cartoon characters, etc. A few users are asked to use our system to gather relevance feedback information into the semantic matrix. Some statistical figures to describe the initial semantic matrix are shown below:

- The total number of positive feedbacks (i.e.,  $\sum_{i,j} m_{ij}$ ) is 17823.
  - The total number of row vectors in  $M$  that are not zero vector is 271.
- Therefore, the average positive feedbacks contained in one row vector are  $\frac{17823}{271} = 66$ .

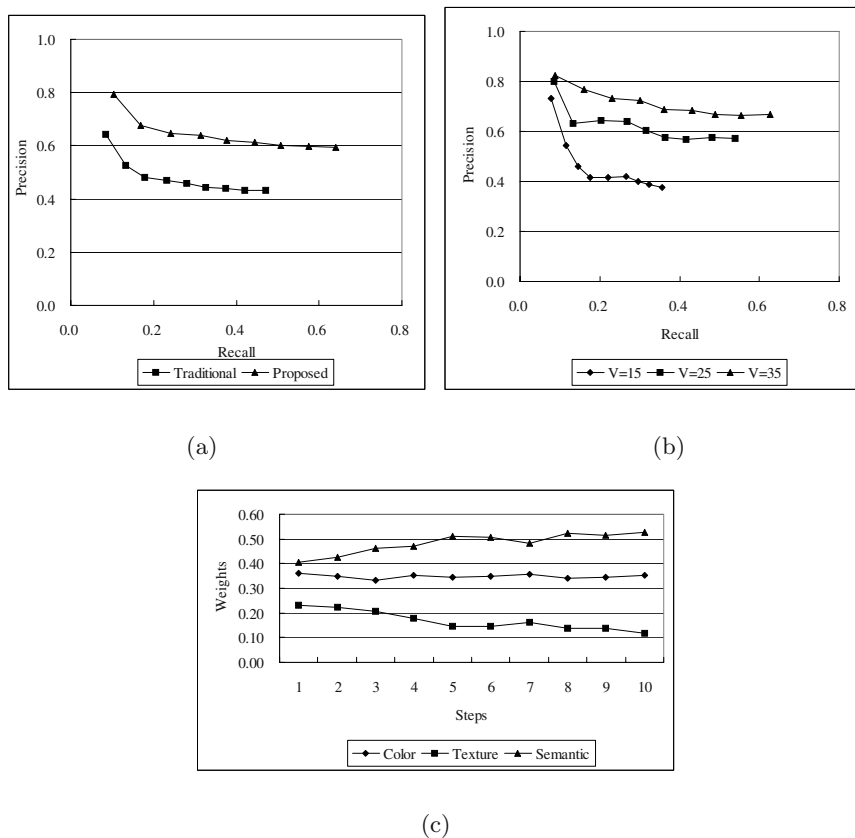
As the size of an image database increases, the semantic matrix also requires a larger storage. It is, however, found that the semantic matrix is actually a kind of sparse matrix, which means that the large portion of the semantic matrix is filled with zero or empty. There are many well-known approaches to represent a sparse matrix in a memory-efficient manner. For instance, one way to relieve the memory requirement is to maintain only the non-zero elements for each row vector in the semantic matrix.

### 3.2 Experimental Results

Figure 3(a) compares the precision-recall graphs of the traditional RF method and the proposed RF method, where the nine values are obtained by varying the number of resultant images, such as 9, 18, 27,  $\dots$ , 81. Here, the traditional



method employs only two low-level features (i.e., color and texture) while the proposed method uses the high-level semantic feature as well as the two low-level features. The above precision-recall graphs are obtained by averaging the search results of any 20 query images. Furthermore, the precision-recall graph of the proposed method is generated using the initial semantic matrix described in the previous section. In terms of search performance, the proposed method seems to be about 20% better than the traditional method.



**Fig. 3.** (a) Precision-recall graphs of the traditional and the proposed RF methods, (b) Transition of precision-recall graph of the proposed method as the relevance feedback information keeps being added into the semantic matrix, and (c) Variation of three weights during a search session.

Figure 3(b) shows how the precision-recall graph of the proposed method changes as the semantic matrix grows. The symbol  $V$  in the legend indicates the average amount of relevance information contained in a row vector in the semantic matrix. The greater the value of  $V$ , the greater the accumulated infor-

mation in the semantic matrix. As we expect, the search performance continues to improve as the value  $V$  increases.

Figure 3(c) is another chart to demonstrate the transition of three weights during a search session. As the search session goes on, the weight of the semantic feature tends to increase while the weights of color or texture decrease or remain unchanged. This observation tells that the semantic feature plays a critical role in finding relevant images quickly. False positive rates can be also dramatically reduced thanks to such favorable behavior of the semantic feature.

## 4 Conclusions

Relevance feedback (RF) is a mechanism introduced earlier to exploit a user's perceptual feedback in image retrieval. Though RF is an intriguing concept for interactive image retrieval, it has one serious drawback, which is that RF ignores valuable feedback information generated from user interactions during search sessions.

In this paper, we propose a novel RF mechanism strengthened with a capability to store and reuse the relevance feedback information effectively. Specifically, the proposed method constructs a semantic space for a large collection of images by accumulating human perceptual responses over time through relevance feedback, and dynamically combines the accumulated high-level relevance information with low-level features to further improve the retrieval effectiveness.

Experimental results show that the retrieval performance of the proposed method is greatly enhanced compared with traditional RF methods and gets better and better as time passes. Furthermore, it is discovered that the semantic feature plays a critical role in finding relevant images quickly and reducing the false positive rates

## References

1. Yong Rui, Thomas S. Huang, Michael Ortega, and Sharad Mehrotra, "Relevance Feedback: A Power Tool in Interactive Content-Based Image Retrieval", in IEEE Tran on Circuits and Systems for Video Technology, Special Issue on Segmentation, Description, and Retrieval of Video Content, pp. 644-655, Vol. 8, No. 5, Sept, 1998.
2. I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos, "The Bayesian image retrieval system, PicHunter:theory, implementation, and psychophysical experiments," in IEEE Transaction on Image Processing, Vol. 9, pp. 20-37, Jan 2000.
3. H. Muller, W. Muller, S. Marchand-Maillet, and T. Pun, "Strategies for positive and negative relevance feedback in image retrieval," in Proc. of IEEE Conference on Pattern Recognition, Vol. 1, pp. 1043-1046, Sep 2000.
4. Jing Huang, S. Ravi Kumar, Mandar Mitra, Wei-Jing Zhu and Ramin Zabih. "Image Indexing Using Color Correlograms," in IEEE Conference on Computer Vision and Pattern Recognition, pp. 762-768, June 1997.
5. Seong-O Shim and Tae-Sun Choi, "Edge color histogram for image retrieval", 2002 Int'l conf. on Image processing, pp. 957-960, vol. 3, Jun.2002.

# Faster Exact Histogram Intersection on Large Data Collections Using Inverted VA-Files

Wolfgang Müller and Andreas Henrich

Universität Bayreuth, 95448 Bayreuth, Germany,  
Wolfgang.Mueller2@uni-bayreuth.de,  
<http://ai1.inf.uni-bayreuth.de/mitarbeiter>

**Abstract.** Most indexing structures for high-dimensional vectors used in multimedia retrieval today rely on determining the importance of each vector component at indexing time in order to create the index. However for Histogram Intersection and other important distance measures this is not possible because the importance of vector components depends on the query. We present an indexing structure inspired by VA-file and Inverted file that does not need to determine the importance at indexing time in order to perform well. Instead, our structure adapts to the importance of vector components at query processing time.

Success of this approach is demonstrated in experiments on feature data extracted from a large image collection.

## 1 Introduction

The research area of Content Based Image Retrieval (CBIR) investigates methods to help users find images in large collections.

Interaction modes and details vary, but most CBIR (and related) applications are driven by (a) a *feature extraction* that transforms each image of the collection into a real valued multidimensional feature vector, (b) a *distance measure or a similarity measure* that permits evaluating the similarity of a pair of images, (c) an *indexing structure* that permits rapid  $k$ -NN ( $k$ -nearest neighbor) searches in the high-dimensional vector space, as well as (d) a *learning method* that permits learning from user (*e.g.* relevance) feedback during use of the CBIR system.

Unfortunately, most publications about CBIR treat the CBIR specific issues ((a), (b), and (d)), seeing indexing structures (c) just from the user's perspective. This is not helped by the fact that literature about indexing structures is typically uniquely using Euclidean distance, not making sure that the needs of CBIR and related fields are met (*i.e.* they are treating just (c)). Few publications address the relation between indexing and feedback in CBIR (for example [6,7]). The present paper intends to be one of these few. It is about a disk-based indexing structure adapted to CBIR-specific distance measures.

In most CBIR systems, as in this paper, the image collection is viewed as a large collection of data points in a high-dimensional real-valued vector space. When performing a query by visual example (QBvE), a query image  $\mathbf{q}$  is a point in that space, and we are looking for the  $k$  data points  $\mathbf{r}_i$  within the collection

closest to the query. Closeness between data points  $\mathbf{r}_1, \mathbf{r}_2$  is defined via a distance measure (Distance of  $\mathbf{r}_2$  given  $\mathbf{r}_1$ )  $\Delta(\mathbf{r}_2|\mathbf{r}_1)$ <sup>1</sup>.

Most indexing structures adapted to high-dimensional vectors known to us perform well in the scenario where the distance measure  $\Delta(\cdot|\cdot)$  is chosen once, when indexing the data. In particular, it is assumed that the discriminative power of each vector component can be determined at indexing time. A successful example of such an indexing structure is the VA-file (vector approximation file, [8]) for  $k$ -NN queries on real-valued vectors. It uses vector approximations for filtering candidates, *i.e.* vectors. The full precision values of all candidates will then be tested in order to find out if each candidate is within the  $k$ -NNs. However, for some popular CBIR distance measures, (*e.g.* Histogram Intersection, Kullback-Leibler Divergence, Cosine Distance) the discriminative power of a vector component varies depending on the query. We will call the discriminative power of a vector component henceforth the *Importance* of a vector component, to be defined more formally below.

While in VA-files the quality of the approximation is fixed at indexing time for each vector component, *our contribution is an indexing structure that uses the importance of vector components for choosing the quality of the approximation at every query.* Our structure, the Inverted Vector Approximation (IVA) file combines the advantages of the inverted file (a well-known data structure from information retrieval, also used in CBIR [6]) with the advantages of the VA-file.

This paper is organized as follows: in the next section we describe the VA-file. Then we describe the query model for which IVA-files are optimized, as well as the notion of vector component importance that is used for choosing the appropriate approximation level (section 3). Then, in section 4, we describe Inverted VA files as VA-files with changed storage and evaluation order. In section 5 we give a method to provide diverse levels of approximation for components of color histograms. Lastly, we present experiments, related work and future work.

## 2 VA-Files

The VA-file is motivated by the observation that the *curse of dimensionality* makes it impossible for tree-based multi-dimensional indexing structures (*e.g.* [4] and others) to obtain better than linear complexity for  $k$ -NN search in spaces with dimensionality  $n > 10$  [8]: Tree-based indexing structures have to consider  $c \cdot N$  (with  $c \approx 1$ ) vectors when querying a collection of  $N$  vectors. The basic idea of the VA-file is to accept this as a fact, and to accept linear complexity, but to minimize the overhead in order to be faster than tree-based indexing structures. This is achieved in a two-pass process.

<sup>1</sup> For brevity, we do not introduce an additional notation for *similarity measures*. The difference between distance and similarity measures is that for distances smaller values indicate better match whereas for similarity measures higher values indicate better match. We always write  $\Delta$  and  $\Delta(y|x) < \Delta(z|x)$ , thinking in terms of distances, *i.e.*  $x$  matches  $y$  better than  $z$ .

The base assumption in optimizing the VA-file's performance is that the time complexity of one query is determined by the number of data blocks that have to be read from disk during the query. If for processing the query,  $c \cdot N$  vectors need to be looked at, then the number of data blocks to be read during a query process is proportional to the total number of bits needed for representing a data vector within the context of our query.

Typical tree-based indexing structures store the data vectors in full precision. Each vector component corresponds to a 32 bit `float` value. The VA-file however, stores each vector twice: once in full (`float`) precision, once its approximation ( $b \ll 32$  bits per component). In the approximation each  $b$ -bit value designates a float interval.

A  $k$ -NN query is processed in a two-pass process: in the first pass, approximations are used for obtaining so called *candidates*, *i.e.* data points for which we cannot rule out the possibility that they are among the query result (*i.e.* the  $k$ -NN) by looking at the approximation alone. For some of the candidates we need to look at the full precision vectors, possibly discarding more candidates from the candidate set while we are finding more and more of the  $k$  nearest neighbors, because we get new bounds for being among the  $k$ -NN during this process.

The performance of the VA-file is determined by the following factors:

**Number of bits needed per approximation component:** As *all* approximations are read during the first pass of the query processing, the number of blocks read in this phase is proportional to the number of bits to be read per vector component.

**Size of the candidate set:** Of course, the number of candidates influences how many data vectors have to be read in full precision. In fact, as hard disks are block-oriented devices, we will have to read statistically *almost one entire block* per full precision data vector we look at.

**Block size:** The data block that needs to be read to verify the candidate status of a data vector will possibly contain a large number of data vectors ( $\approx 10$  vectors) that are not member of the candidate set, *i.e.* that are not interesting for our query process. Although these data points will not be considered, they are still read, and still are part of the cost of the query process.

Note that the number of bits needed per component approximation and the size of the candidate set are directly related to the quality of the approximation. There is some literature on choosing useful approximations for skewed distributions [3,9]. Both methods, as well as the initial VA-file approximation method have in common that the approximation is fixed at indexing time. There is no possibility to adapt approximation quality to the query without updating the index. That is, the methods adapt to the data, however costly and slowly.

### 3 Queries and Component Importance

What we need to do when processing a query is to find the  $k$  vectors  $\mathbf{r}_l$ ,  $l \in \{1, \dots, k\}$  closest with respect to a query  $\mathbf{q}$  from which a distance measure

$\Delta(\cdot|\mathbf{q})$  (given the query) is inferred. We require that the distance measure can be expressed as a component wise sum  $\Delta(\mathbf{r}_i|\mathbf{q}) = \sum_{j=1}^n \Delta_j(r_{i,j}|\mathbf{q})$ .

Now, let us imagine for an instant, we would like to perform an inexact  $k$ -NN query on our data collection. When calculating each of the distances  $\Delta(\mathbf{r}_i|\mathbf{q})$  we would not evaluate the distance for all vector components  $\sum_{j=1}^n \Delta_j(r_{i,j}|\mathbf{q})$ , but rather only for a subset  $J \subset \{1, \dots, n\}$  of vector components  $\sum_{j \in J} \Delta_j(r_{i,j}|\mathbf{q})$ . It is intuitively evident that for two sets of components  $J_1 \subset \{1, \dots, n\}$  and  $J_2 \subset \{1, \dots, n\}$  the result will differ most of the time if  $J_1 \neq J_2$ . Moreover, we will assume that ignoring some components will change the results more strongly with respect to the results of the full evaluation than ignoring other ones.

This is what importance of vector components is about. An important component is a vector component that, if not evaluated, changes the ranking strongly. We define the importance of vector component  $j$  given the query  $\mathbf{q}$  as:

$$I(j|\mathbf{q}) = \max_{i=1}^N (\Delta_j(r_{i,j}|\mathbf{q})) - \min_{i=1}^N (\Delta_j(r_{i,j}|\mathbf{q})) \quad (1)$$

That is, the Importance of component  $j$  is given by the variability of its contribution to  $\Delta$ , *i.e.* its discriminative power.

The implementation of VA-files involves the approximation of vector components. That is, we evaluate the distance for all components, but the evaluation is not exact. The concept of component importance is also important here. It allows us to choose how precise or how coarse we can make the approximation of a given component. Components of high importance will be better approximated than those of low importance. However, it is important here to note that neither the VA-file nor our variant, the IVA-file, perform inexact  $k$ -NN queries, because the inexact first pass is supplemented by the second pass exactly checking the candidates.

In this publication we are interested in non-Gaussian data distributions and distance measures other than Euclidean distance.<sup>2</sup> The definition of importance we gave in Eq. 1 proves very useful in these cases, as the following example shows:

Assume documents represented as  $n$ -bin histograms, and assume they are going to be ranked using histogram intersection  $\Delta(\mathbf{r}|\mathbf{q}) := \sum_{j=1}^n \min(q_j, r_j)$

Let histogram bin (*i.e.* vector component) number 42 represent the color “purple”. Furthermore, assume the query  $\mathbf{q}^{42}$  to be a vector with  $q_j^{42} = 1$  for  $j = 42$  and  $q_j^{42} = 0$  otherwise.  $\mathbf{q}^{42}$  would correspond to a uniformly purple query image. In this case  $I(j|\mathbf{q}^{42}) = I_{42}$  for  $j = 42$  and  $I(j|\mathbf{q}^{42}) = 0$  otherwise, for some constant  $I_{42} > 0$  whose exact value does not matter here.

What’s relevant for us here is that when a vector  $\mathbf{r}_i$  is ranked with respect to a uniformly colored image:  $\mathbf{q}^{42}$ , no histogram components matter at all (*i.e.*

<sup>2</sup> In the case of Gaussian distribution of data points and Euclidean distance the way to quantify and use importance is to perform a principal components analysis (PCA), *i.e.* finding a basis of the vector space that consists of the Eigenvectors of the covariance matrix. In this case, the least important vector components are those corresponding to the Eigenvectors with the smallest Eigenvalues. VA<sup>+</sup>-files [3] exploit PCA in order to reduce the size of the approximation.

they have *zero importance*) except the one that is present in the query image. Indeed evaluating just  $\Delta_{42}(\cdot|q_{42}^{42})$  suffices for obtaining the correct  $k$ -NN query result, because  $\Delta_{42}(r_{i,42}|\mathbf{q}^{42}) = \Delta(\mathbf{r}_i|\mathbf{q}^{42})$  for all  $\mathbf{r}_i$ .

In contrast, a PCA-based method would evaluate the importance of vector components based on their variance over the whole collection. As a consequence, a component  $c$ ,  $c \neq 42$  could be assigned a larger importance than component 42 due to the large variance of  $r_{i,c}$ . Clearly this is not useful in our example, as the importance of vector components is solely determined by the query.

The above example is clearly extreme. However, also in realistic settings with color histograms calculated from real data collections, the essential ingredient stays the same: the importance of the vector components depends on the distance measure  $\Delta$ , the query  $\mathbf{q}$  and the data collection  $\mathcal{C}$ . This is, why we write the importance of component  $j$  for query  $\mathbf{q}$  as  $I(j|\mathbf{q}, \mathcal{C}, \Delta)$  in the following.

## 4 From VA to Inverted VA

In VA-files, vector approximations are stored and read “line by line”, *i.e.* document by document, vector by vector. It is not possible to read partial vectors because hard disks are block-oriented devices: typically, the approximations for a number of vectors will fit into one block. Reading only parts of one block won’t bring any efficiency gain. This observation leads us to changing the storage and evaluation order of the VA-file and then looking at the possible benefits.

*Essentially, Inverted VA-files are VA-files with a changed storage and evaluation order.* In IVA-files, vector approximations are stored and read column by column. That is, when processing a query, the query processor first reads and processes the first components  $r_{i,1}$  for each data point  $\mathbf{r}_i$ , before then processing the second components  $r_{i,2}$ , and so forth. The second phase of IVA-queries is the same as for queries using VA-files.

While this might appear only a minor change, this change of storage order allows us to *choose which components to read*. We use this newly-acquired liberty of what to read as follows: We store each component of each data vector  $\mathbf{r}_i$  at several levels of approximation quality. On processing a query we choose for each component the level of approximation quality needed. After having chosen, we will read the corresponding column from the IVA-file.

## 5 Approximations for Histogram Intersection

Now the only ingredient missing for successful use of the IVA-file for color histograms is the approximation itself. The approximation is simply an array mapping of a  $b$  bit integer to  $2^b$  real-valued intervals. While the methods presented up to this point in the paper have been generic, now we will concentrate on the histogram intersection distance.

Let us come back to our example given in section 2. Here, we showed that the importance is zero for each component for which the query component  $q_j$

is zero. More generally, the importance of the  $j$ -th component is limited by the value of  $q_j$ :  $I(j|\mathbf{q}) = \max_{i=1}^N (\min(q_j, r_{i,j})) - \min_{i=1}^N (\min(q_j, r_{i,j}))$ .

In other words, for a given document  $\mathbf{r}_i$ , the contribution of component  $j$  to  $\Delta(\mathbf{q}, \mathbf{r}_i)$ ,  $\Delta(q_j, r_{i,j}) := \min(q_j, r_{i,j})$  is limited by  $q_j$ . We can use this fact for the approximation of  $\mathbf{r}_i$ .

For generating the approximations that performed best in our experiments (we also tried the ones suggested initially for Euclidean distance in [3]), we chose a small number  $\beta$  and cut the interval between minimum ( $r_{-,j} := \min_i r_{i,j}$ ) and maximum ( $r_{+,j} := \max_i r_{i,j} + \epsilon$ ) value of a component within the collection into  $2^\beta$  non-overlapping equal-sized chunks, obtaining a set of intervals ( $R$  abbreviates “Region”, and  $0 \leq \ell < 2^\beta$ ):

$$\begin{aligned} R(\ell, \beta, \beta, r_{+,j}, r_{-,j}) &:= \left[ r_{-,j} + \frac{\ell}{2^\beta} \cdot (r_{+,j} - r_{-,j}), r_{-,j} + \frac{\ell+1}{2^\beta} \cdot (r_{+,j} - r_{-,j}) \right) 2 \\ &:= [R_{\min}(\ell, \beta, \beta, r_{+,j}, r_{-,j}), R_{\max}(\ell, \beta, \beta, r_{+,j}, r_{-,j})] \end{aligned} \quad (3)$$

Eq. 3 is just shorthand for Eq. 2. We now define for  $b < \beta$  bits:

$$R(\ell, b, \beta, r_{+,j}, r_{-,j}) := \begin{cases} R(\ell, b, \beta, r_{+,j}, r_{-,j}) & : \ell < 2^b - 1 \\ \left[ r_{-,j} + \frac{2^b - 1}{2^\beta} \cdot (r_{+,j} - r_{-,j}), r_{+,j} \right) & : \text{otherwise} \end{cases} \quad (4)$$

That is, the first  $2^b - 1$  intervals are exactly the same as with  $\beta > b$ , and the remaining interval covers the complete rest.

*Now, how do we choose the level of approximation?* Given a query  $\mathbf{q}$  we can choose the minimal  $b$  such that for all  $\ell$  the  $b$ -bit approximation produces the same histogram intersection values as the  $\beta$ -bit approximation:

$$\min(q_i, R_{\min}(\ell, b, \beta, r_{+,j}, r_{-,j})) = \min(q_i, R_{\min}(\ell, \beta, \beta, r_{+,j}, r_{-,j})) \quad (5)$$

$$\wedge \min(q_i, R_{\max}(\ell, b, \beta, r_{+,j}, r_{-,j})) = \min(q_i, R_{\max}(\ell, \beta, \beta, r_{+,j}, r_{-,j})) \quad (6)$$

This method works already fairly well, however we can improve on this by relaxing the constraint a bit. We choose a minimal  $b$  such that

$$\min(q_i, R_{\max}(\ell, b, \beta, r_{+,j}, r_{-,j})) - \min(q_i, R_{\min}(\ell, b, \beta, r_{+,j}, r_{-,j})) \leq \frac{(r_{+,j} - r_{-,j})}{2^\beta} \quad (7)$$

This makes the  $b$ -bit approximation (for some queries and documents) slightly worse than the  $\beta$ -bit approximation. However, this increases the number of times, where we do not have to read any approximation at all ( $b = 0$ -bit approximation).

**Example:** Let us continue our example already given in section 2. Let us assume that the minimum and maximum values of the 42nd component within the data collection are 0 and 1, respectively ( $\min_{i=1}^N r_{i,42} = 0$  and  $\max_{i=1}^N r_{i,42} = 1$ ), and let us choose  $\beta = 5$ .

With this choice,

$$R(\ell, 5, 5, 1, 0) = \left[ \frac{\ell}{32}, \frac{\ell+1}{32} \right) \quad (8)$$

$$R(\ell, b, 5, 1, 0) := \begin{cases} \left[ \frac{\ell}{32}, \frac{\ell+1}{32} \right) & : \ell < 2^b - 1 \\ \left[ \frac{\ell}{32}, 1 \right) & : \text{otherwise} \end{cases} \quad (9)$$



We now choose values for  $q_{42}$  and give the corresponding  $b$  to be used for the approximation. We consider three cases:

- (1) If  $q_{42} = 1$ , evidently, we need  $b = \beta = 5$ .
- (2) If  $q_{42} = 0$ , then  $b = 0$  yields  $[0, 1)$  as only “approximation interval”. For all  $r_{i,42}$  within the collection

$$\begin{aligned} \min(0, 0) &= \min(q_{42}, R_{\min(\ell, \underline{5}, 5, 1, 0)}) = \min(q_{42}, R_{\min(\ell, 0, 5, 1, 0)}) = \min(0, 0) \\ \wedge \min\left(0, \frac{1}{32}\right) &= \min(q_{42}, R_{\max(\ell, \underline{5}, 5, 1, 0)}) = \min(q_{42}, R_{\max(\ell, 0, 5, 1, 0)}) = \min(0, 1) \end{aligned}$$

So, we can choose  $b = 0$  without degrading approximation quality.

- (3) If  $q_{42} = \frac{7}{64}$ ,  $b = 2$  yields  $[0, \frac{1}{32})$ ,  $[\frac{1}{32}, \frac{1}{16})$ ,  $[\frac{1}{16}, \frac{3}{32})$ ,  $[\frac{3}{32}, 1)$  as only approximation intervals.

Now, for all  $\ell = 0, 1, 2$  the resulting bounds will be exactly equal to those of the  $b = \beta = 5$  bit approximation. So for  $\ell = 0, 1, 2, 3$  a 2-bit approximation fulfills Eq. 5, Eq. 6. The interesting case is  $\ell \geq 4$ . Let us choose  $\ell = 4$ . We find:

$$\begin{aligned} \frac{7}{64} &= \min(q_{42}, R_{\min(4, \underline{5}, 5, 1, 0)}) \neq \min(q_{42}, R_{\min(4, 2, 5, 1, 0)}) = \frac{3}{32} \\ \frac{7}{64} &= \min(q_{42}, R_{\max(4, \underline{5}, 5, 1, 0)}) = \min(q_{42}, R_{\max(4, 2, 5, 1, 0)}) = \frac{7}{64} \end{aligned}$$

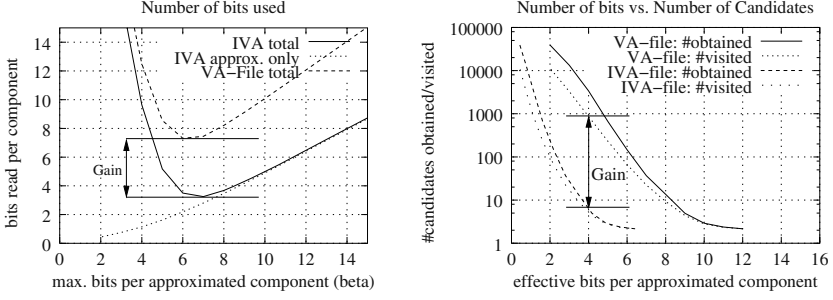
That is, for  $\ell = 4$ , the approximation condition given in Eq. 5 is *not* fulfilled *i.e.* the approximation quality when using  $b = 2$  bits is slightly degraded with respect to using  $\beta = 5$ -bit approximations in this case. However, Eq. 7 holds here, *i.e.* when using the second way of approximation, described in the previous section, we would use 2 bits. When using  $b = 3$  bits, Eq. 5 and Eq. 6 both hold, *i.e.* when using the first way of approximation described in the previous section, we would use 3 bits.

As this example shows, we have the opportunity to use fewer bits by degrading approximation quality slightly. In our experiments, the loss of approximation quality (*i.e.* the increase in size of the candidate set) was offset by the savings in approximation size.

## 6 Experiments

In the experiments we present here we used a collection of 66616 32-bin color layout histograms (*i.e.* color histograms with 32 components), obtained from [kdd.ics.uci.edu/databases/CorelFeatures/CorelFeatures.html](http://kdd.ics.uci.edu/databases/CorelFeatures/CorelFeatures.html). The site reports that the data has been used in [5]. These histograms have been extracted from the Corel image collection. They are free for scientific use. The 32 bins correspond to 4 levels of hue, and 2 levels of saturation in 4 image regions, capturing both color and layout in the histogram.

For our experiments, we varied  $\beta$  from 2 to 12 bits. Each time, we did 100 1-NN test queries using a VA-file and an IVA-file, respectively, and counted during the test queries the number of 8192-byte blocks read, calculating from the



**Fig. 1.** Overall performance of 1-NN query on VA-file. On the left, the horizontal lines within the plot highlight the value that matters: the overall number of bits read per component, and the gain obtained by use of the IVA-file. Here, the IVA-file performs about 2 times better than the VA-file. On the right, we show that the number of candidates visited for an average approximation bit width of  $b = 4$  is by two orders of magnitude smaller for the IVA-file than for the VA-file.

number of blocks the effective number of bits read. The average number of bits read per vector component on the test queries for one given  $\beta$  is one data point in our plots. The choice of  $\beta$  influences the precision of the approximation, and thus the number of candidates. The total number of bits read depends on  $\beta$ , and the number of candidates for which the exact vector has to be read, so we need a tradeoff for which the average overall number of bits read becomes minimal. The number of bits read per component at this minimum is the parameter that determines the performance of the VA-file or IVA-file. The comparative plot obtained using an IVA-file and a VA-file are shown on the left side of Fig. 1. The overall number of bits read per vector component is more than 2 times better for the IVA-file than the number of bits read using the VA-file.

To highlight the increase in the quality of the approximation for a given number of bits effectively used, [3,9] plot the number of candidates obtained in the first (*i.e.* approximation) pass and the number of candidates actually visited versus the number of bits actually used in the approximation (average  $b$  plus overhead for blocking). The right side of Fig. 1 shows such a plot for our test data. We see two pairs of curves. Each pair depicts the number of candidates generated by the first pass, and (a little lower) the number of candidates that actually have to be visited. The pair of curves depicting the IVA-file is distinctly lower than that depicting the VA-file. In particular one can see that for a given number of bits per approximation (average  $b$ ), the IVA-file needs to visit up to two orders of magnitude fewer candidates than the corresponding VA-file.

## 7 Related Work

There are a few groups that have worked with structures similar to the IVA-file. Squire *et al.* [6] used search pruning on inverted files to improve response time.

However, the process was a one-pass process yielding *approximately* the same result as full evaluation of the query. De Vries *et al.* [2] use full precision *Vertically Decomposed Data* (*i.e.* column vectors) in order to choose what parts of the query to evaluate as part of a two-pass query process. Here each tuple is read in full precision. However, only some (not all) tuples are read. They report savings of about  $\frac{2}{3}$  rds with respect to full evaluation (compared to savings of about  $\frac{8}{9}$  th reported in our experiments). The motivation for vertically decomposed data and pruning is cheap update. Lastly, Böhm *et al.* [1] report experiments in which they index feature data belonging to each group of features (color, texture *etc.*) in its own VA-file, merging the results of queries on each file.

## 8 Further Work

In the future, we want to explore the usefulness of our data structure for other distance measures as *e.g.* the cosine distance and the Kullback-Leibler Divergence. In addition to that we want to show the usefulness of our method also for other distance measures, if relevance feedback changes the importance of vector components at query time.

## References

1. K. Böhm, M. Milvoncic, H.-J. Schek, and R. Weber. Fast Evaluation Techniques for Complex Similarity Queries. In *Proc. Intl. Conf. on VLDB*, 2001.
2. A. P. de Vries, N. Mamoulis, N. Nes, and M. L. Kersten. Efficient k-NN Search on Vertically Decomposed Data. In *Proc. SIGMOD*, Madison, WI, USA, June 2002.
3. H. Ferhatosmanoglu, E. Tuncel, D. Agrawal, and A. E. Abbadi. Vector approximation based indexing for non-uniform high dimensional data sets. In *CIKM: ACM Intl. Conf. on Information and Knowledge Management*. McLean, VA, USA, 2000.
4. A. Guttman. R-trees: A dynamic index structure for spatial searching. In B. Yormark, editor, *SIGMOD'84, Proc. of Annual Meeting, Boston, MA, 1984*.
5. M. Ortega, Y. Rui, K. Chakrabarti, K. Porkaew, S. Mehrotra, and T. S. Huang. Supporting ranked boolean similarity queries in MARS. *IEEE Transactions on Knowledge and Data Engineering*, 10(6), December 1998.
6. D. M. Squire, H. Müller, and W. Müller. Improving response time by search pruning in a content-based image retrieval system, using inverted file techniques. In *IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'99)*, Fort Collins, CO, USA, 1999.
7. J. Tesic and B. S. Manjunath. Nearest Neighbor Search for Relevance Feedback. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, Madison, WI, USA, June 2003.
8. R. Weber, H.-J. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *Proc. Intl. Conf. on VLDB*, New York, USA, 1998.
9. P. Wu, B. Manjunath, and S. Chandrasekaran. An Adaptive Index Structure for High-dimensional Similarity Search. In *Proc. IEEE Pacific-Rim Conf. on Multimedia, Advances in Multimedia Information (PCM '01)*, Beijing, China, 2001.

# A Graph Edit Distance Based on Node Merging

S. Berretti, A. Del Bimbo, and P. Pala

Dipartimento Sistemi e Informatica  
Università di Firenze  
via S.Marta 3, 50139 Firenze, Italy  
{berretti,delbimbo,pala}@dsi.unifi.it

**Abstract.** In this paper a novel solution is proposed for error tolerant graph matching. The solution belongs to the class of edit distance based techniques. In particular, the original edit distance based framework is extended so as to account for a new operator to support node merging during the matching process.

An analysis of the computational complexity of the proposed solution is presented as well as some experimental results about its application to retrieval by content of color images.

## 1 Introduction

Description of visual objects and computation of their similarity is a key issue in many different application contexts. These include, and are not limited to, surveillance, audio/image/video/3D information retrieval and object recognition to say a few.

Several data structures can be used to organize information about object features, and among these structures, graphs are the most versatile and powerful. The use of graphs to represent object features typically encompasses segmentation of the object into parts and use of graph nodes to represent features of object parts and graph edges to represent their relationships. For this purpose, nodes and edges are associated (labelled) with additional information to describe the characteristics of object parts and of their relationships, respectively. Graphs enable an accurate description of object parts and of their relationships, and an accurate description is usually the first step to achieve an accurate measure of similarity between objects.

Determining the similarity between two graphs is usually referred to as *Graph Matching*. For an updated review of methods and techniques related to graph matching, the interested reader can refer to [1]. Early approaches to graph matching addressed the problem of *exact* matching. Under this perspective, the solution to the graph matching problem is to find a *graph isomorphism*, that is a bijective function that associates with every node/edge in the first graph one node/edge in the second one. It is assumed that the two graphs have the same number of nodes/edges, the same labels and the same edge structure. However, the process of eliciting graphs from raw data is usually affected by noise and errors of various types. As an example, the segmentation of an image can be considered. Typically, many regions that are detected don't correspond to the regions that would be outlined by a manual annotator: the image may be either over- or under-segmented in some parts. As a result, two graph representations of the same or very similar objects may be different.

A first solution for matching graphs that are not identical is to address *subgraph* rather than graph isomorphism: a subgraphs isomorphism between two graphs being an isomorphism between one of the two graphs and a subgraph of the second one. In classical subgraphs isomorphism methods [2], [7] the best match is found using the  $A^*$  search method. This is guaranteed to find the optimal match, but may require exponential time due to the NP-completeness of the problem. In [13], a system is presented that exploits subgraph matching to support retrieval of graphic logos from a database of color images taken from advertisements and magazines. The system is translation and scale invariant and can accurately locate a query logo in a target image. However, its ability to retrieve similar objects in addition to identical ones is very limited.

In order to cope effectively with comparison of similar but not identical objects, the concept of *inexact* matching should be considered [4], [5], [6]. Only the adoption of inexact graph matching techniques can enable accurate and effective measure of similarity between objects that are visually similar though not identical. It should be considered that the measure of similarity is not only intended to model differences between two objects in terms of translation and scale. Rather, the ability of a system to capture the similarity between two objects is useful for comparison of objects that are different to begin with, or that become different due to noise, distortions or image processing operations (e.g. segmentation).

In the context of inexact graph matching, an innovative approach is based on the notion of *graph edit distance* [8]. This is defined with respect to a set of edit operations—delete, insert, rename—that can be applied to nodes and edges of the first graph in order to match it against the second one. Each edit operation is associated with a cost. In this way, the overall effect of all the edit operations that are applied to one graph can be quantified through an overall cost that sums up the costs associated with individual operations. The higher the cost of the operations that are applied the more dissimilar the two graphs.

In this paper a novel solution is proposed for error tolerant graph matching. The solution belongs to the class of edit distance based techniques. In particular, the original edit distance based framework is extended so as to account for a new operator to support node merging during the matching process. It should be considered that in the context of edit distance based techniques, node merging is not equivalent to a sequence of node deletion and insertion. Indeed, the graph that results from the application of a node merging operation can also be obtained through the application of appropriate deletion and insertion operations. However, the cost that is associated to the two transformations is not the same: in the general case, the cost of one operation (merging) is less than the cost of two operations (deletion and insertion). Furthermore, the merging operation should be associated with a much lower cost than deletion and insertion as the former condense in one node the information scattered in two or more nodes whilst the latter two either remove or add new information to the graph.

Techniques for graph matching based on node splitting and merging have been previously used for object tracking [10] and image content description [3], [9], [11]. However, in the proposed solution, instead of applying edit operations only to one of the two graphs, graph matching is achieved by editing both graphs. In this way, the application of edit operation is equivalent to a process by which the two graphs evolve toward a common graph structure.

The paper is organized as follows: in Sect.2 the graph matching problem is formally stated with reference to the new operator of node merging. The algorithmic implementation and an estimate of its computational complexity are discussed in Sect.3. Some experimental results are reported in Sect.4 and conclusions are drawn in Sect.5.

## 2 Graph Matching by Node Merging

In the following, a graph is represented (through the same formalism used in [3]) in the form  $g = (V, E, \alpha, \beta)$ , being  $V$  a set of nodes,  $E \subseteq V \times V$  a set of edges,  $\alpha : V \mapsto L_V$  a node labelling function,  $\beta : E \mapsto L_E$  an edge labelling function.  $L_V$  and  $L_E$  are the set of nodes and edge labels, the term label referring to a generic descriptor representing a bunch of information associated with the node/edge. In this sense, a label may be a symbolic descriptor as well as a feature vector retaining prominent characteristics of the part of object associated with the node/edge.

Given a generic graph  $g = (V, E, \alpha, \beta)$ , a subset  $W \subseteq V$  of its nodes is *connected* if for each pair of nodes of  $i, j \in W$  there is a path in  $W$  leading from  $i$  to  $j$ .

In traditional graph matching based on the edit distance, a set of edit operations is defined to transform one graph into another one. The set of edit operations is composed of deletion, insertion and substitution—the effect of this latter operation being a change of label value. In the proposed solution, this set is augmented by the *node merging* operation that replaces a set of connected nodes with one node. The new node replaces the old ones and inherits their properties. In particular, the label that is associated with the new node is computed according to a *feature propagation function*  $F_{pf}$  that accounts for labels associated with replaced nodes.

In order to account for node merging, the matching process is organized as an iterative process that performs the following actions, at each iteration step:

- For each graph node compute the set of *compatible nodes*.
- For each graph node compute the set of *virtual nodes*. This is defined as the set originated by merging the current node with one or more adjacent nodes.
- Compare nodes of the two graphs so as to decide which combinations of nodes should be actually fused.

Detailed information about each iteration step is provided in the following.

### 2.1 Compatible Nodes

Each node is associated with a label that represents information about node features. We assume that a dissimilarity metric is available that enables comparison of node labels so as to derive the dissimilarity between two nodes. In our case, this dissimilarity metric is in the form of a weighted Euclidean distance.

Let  $g = (V, E, \alpha, \beta)$  be a graph,  $i, j \in V$  two nodes and  $\alpha(i), \alpha(j) \in L_V \subseteq \mathbb{R}^n$  their labels. The dissimilarity between nodes  $i$  and  $j$  is measured as:

$$D_\omega(i, j) = [\alpha(i) - \alpha(j)]' \text{diag}(\omega_1, \omega_2, \dots, \omega_n) [\alpha(i) - \alpha(j)] \quad (1)$$

being  $(\omega_1, \omega_2, \dots, \omega_n)$  a set of weights used to balance the relative relevance of node features.

It should be noticed that this definition of node dissimilarity is not restricted to nodes of the same graph: The dissimilarity between nodes of two distinct graphs can be computed prevented that they adopt homogeneous labels (feature vectors).

Given a generic graph  $g = (V, E, \alpha, \beta)$ , two nodes  $i, j \in V$  are *compatible nodes* if both the following conditions hold:  $i$  and  $j$  are adjacent nodes;  $D_\omega(i, j) < \tau_c$ ; being  $\tau_c$  a fixed node compatibility threshold (in the experimental results reported in Sect.4 this threshold was set to  $\tau_c = 0.3$ ).

For a generic node  $i$  of a graph, the set of compatible nodes can be defined. This set is indicated as  $C(i)$  and is composed of the current graph node and all its compatible nodes.

For node  $i$ , the set of compatible nodes  $C(i)$  is used to derive the set of virtual nodes. In general, a virtual node results from the combination (fusion) of one node with one or more compatible nodes. Given a generic node  $i$ , let  $N + 1$  be the cardinality of  $C(i)$ . Let  $C_k^N(i)$  be the set of all  $k$ -combinations of the elements of  $C(i)$  that include node  $i$ . Then, the set of virtual node combinations for node  $i$  is defined as:

$$VN(i) = \{C_k^N(i)\}_{k=0}^N$$

being  $C_0^N(i) = i$ .

Since the cardinality of  $C_k^N(i)$  is  $\binom{N}{k}$ , the cardinality of  $VN(i)$  is  $\sum_{k=0}^N \binom{N}{k} = 2^N$ .

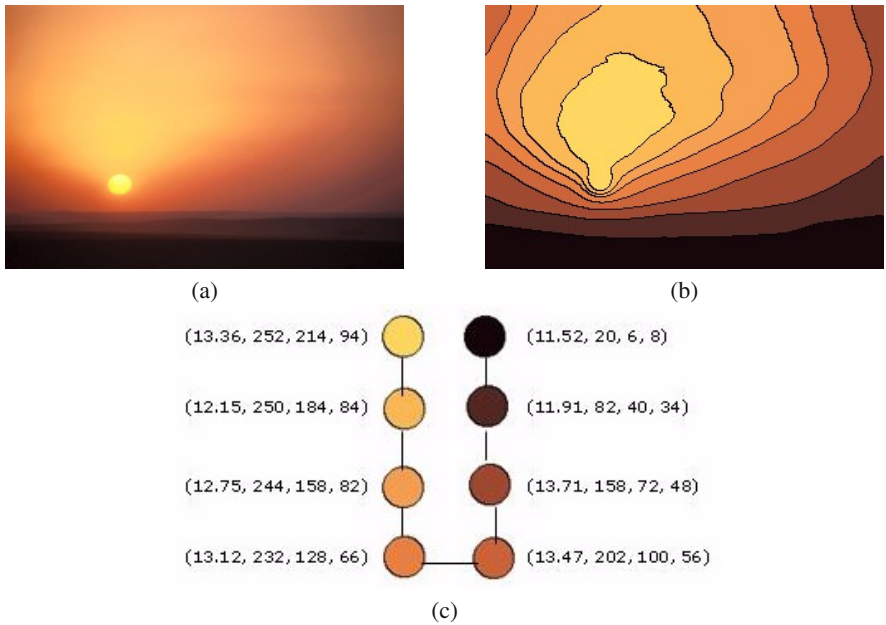
Indeed, each element of  $VN(i)$  is a node. In particular, it can be node  $i$  or any node obtained by merging node  $i$  with one or more compatible nodes. Given a virtual node  $\psi$  the *node originating* function  $\Omega(\psi)$  returns the set of nodes that were merged to originate it. When two or more nodes are merged to create a new node, some criteria must be followed in order to compute the label to assign to the new node. New nodes should inherit information from the nodes from which they originate.

This is accomplished through the definition of a *feature propagation* function  $F_{fp} : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}^n$ . Given a pair of nodes and their labels, this function outputs the label that should be associated with the node originating from the fusion of the first two.

The way in which the feature propagation function acts on the pair of feature vectors associated with the nodes to be merged depends on what is represented in the elements of the feature vector. In general, the feature propagation function may entail ad-hoc knowledge about rules to be applied for each element of the feature vector.

A sample case is shown in Fig. 1. The feature vector of one node combines information about area and color. In particular, the feature vector of node  $i$  is in the form  $\mathbf{f}^i = (f_1^i, f_2^i, f_3^i, f_4^i) \in \mathbb{R}^4$ , being  $f_1^i$  the area of the region represented by node  $i$ , and  $f_2^i, f_3^i, f_4^i$  the three components of its color. In order to combine the feature vectors of two nodes  $i$  and  $j$  the feature propagation function will apply a *summation rule* for the first element of the feature vector and a *mean rule* for the last three elements. That is:

$$F_{fp}(\mathbf{f}^i, \mathbf{f}^j) = (f_1^i + f_1^j, \frac{f_2^i + f_2^j}{2}, \frac{f_3^i + f_3^j}{2}, \frac{f_4^i + f_4^j}{2})$$



**Fig. 1.** (a) A sample image. (b) Segmentation of the image into regions. (c) Graph representation of segmentation results; nodes correspond to regions and are labelled with region area (percentage with respect to image size) and region color (RGB color space).

Virtual nodes that are originated from nodes of the graph in Fig.1 are shown in Fig.2. Each virtual node is evidenced through a dotted contour and is connected to its originating nodes through dotted edges.

## 2.2 Virtual Nodes Comparison

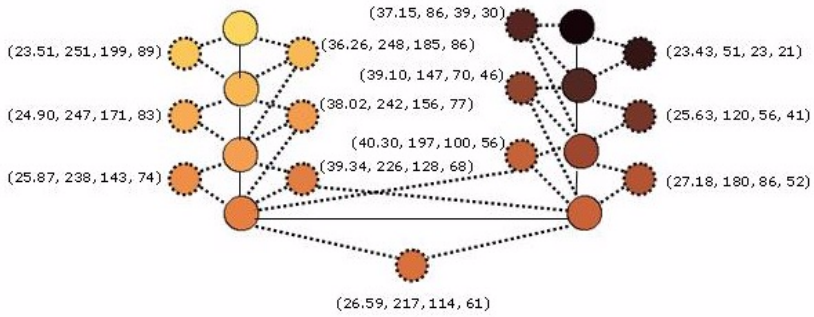
When two graphs are to be compared, for each graph node the virtual node combination set is computed. Virtual node combination sets of the two graphs are compared so as to determine the best node correspondences. In order to find the best node correspondences both actual-to-actual, actual-to-virtual and virtual-to-virtual node comparisons are explored.

Nodes are not compared using the dissimilarity function defined in Eq.(1). Rather, a *context dissimilarity* function  $\mathcal{D}_{cd}(\cdot, \cdot)$  is defined for this purpose. Given two nodes  $i$  and  $j$ , the value of  $\mathcal{D}_{cd}(i, j)$  accounts not only for the dissimilarity of the feature vectors associated with nodes  $i$  and  $j$ , but also for the dissimilarity of their adjacent nodes.

In this way, the purpose of selecting the best node correspondences is twofold: favorite aggregation of nodes that find a counterpart in both graphs; favorite aggregation of nodes that are surrounded by similar nodes (aggregations of nodes).

Comparison of two virtual node combination sets results in the identification of two nodes (belonging to the first and second graphs respectively) that correspond each other. Each one of these two nodes can be either an actual or virtual node. In case it's a virtual





**Fig. 2.** Virtual nodes originated from the graph shown in Fig.1(c). For simplicity, feature vectors associated with original graph nodes are not displayed (they are reported in Fig.1(c)). Each virtual node is evidenced through a dotted contour and is connected to its originating nodes through dotted edges.

node, it becomes an actual node and replaces—in the original graph—all the nodes that originated it.

### 3 Algorithm Implementation and Complexity Analysis

The proposed solution to graph matching requires comparison not only of the original graph nodes but also of the virtual nodes that they originate. Assuming the computational complexity of a traditional subgraph matching problem to be  $O(n^m)$  (being  $n$  and  $m$  the number of nodes of the two graphs), the complexity of the proposed solution scales to  $O((n * \xi)^{(m * \xi)})$  being  $\xi$  the average number of virtual nodes originated from each actual node.

In order to be effectively used for graph comparison, the complexity of the proposed solution needs to be reduced. This is accomplished by adopting a *greedy strategy* for node comparison. According to this strategy, graph comparison is accomplished through an iterative matching process. At each iteration step, the following actions are performed:

- Select one node  $i$  in the first graph and find the most similar node  $j$  in the second one.
- Compute  $VN(i)$  and  $VN(j)$ , that is the virtual nodes originated by nodes  $i$  and  $j$ .
- Compare elements of  $VN(i)$  and  $VN(j)$  to find the best match
- If the best match involves some virtual nodes, e.g. node  $\psi$ , replace all nodes  $\Omega(\psi)$  with  $\psi$ . In the next iteration steps,  $\psi$  is regarded as an actual node (not a virtual one).

Adoption of this greedy strategy for node comparison reduces the computational complexity of the matching process. In fact, node correspondences are found through an iterative exploration of the best possible node mappings and selection of the best mapping at each iteration. However, this approach doesn't guarantee to find the optimal solution to the matching problem.

The computational complexity of the proposed graph matching technique is evaluated as follows. Let us consider two graphs,  $G_1$  and  $G_2$  with  $m_1$  and  $m_2$  nodes, respectively. In addition, without loss of generality, we assume that  $m_1 \leq m_2$ . The matching algorithm is constructed around a main loop which iteratively considers all the nodes in graph  $G_1$  in order to subsequently assign them to nodes in  $G_2$ . According to this, the worst computational complexity can be estimated for the case when, for each node  $i$  in  $G_1$  the following operations are performed:

- (a) Find the node  $j$  in  $G_2$  that is most similar to node  $i$ .
- (b) Build  $VN(i)$  and  $VN(j)$ , that is the sets of virtual nodes for nodes  $i$  and  $j$ .
- (c) Compare  $VN(i)$  and  $VN(j)$  to find the best match between their elements

Complexity of step (a) is  $O(m_2)$ . Assuming that each node has an average number of  $N$  adjacent compatible nodes, the average cardinality of the virtual combination sets is:

$$E[\#VN(i)] = E[\#VN(j)] = \sum_{k=0}^N \binom{N}{k} = 2^N$$

Therefore, the computational complexity of step (c) amounts to  $O(2^{2N})$ . The computational complexity of step (b) is negligible with respect to  $O(2^{2N})$ .

Since steps (a), (b) and (c) have to be performed for each node  $i$  in  $G_1$  the overall computational complexity is  $O(m_1 * (m_2 + 2^{2N}))$ . Hence, management of node merging penalizes the overall complexity of the matching process only when  $m_2 \ll 2^{2N}$ .

## 4 Experimental Results

The proposed approach for graph matching by node merging has been experimented in the application context of image retrieval by visual similarity. In particular, we considered a benchmark database of about 1000 images, representing paintings by different authors, styles and artistic period. Images were initially described by segmenting them into regions according to chromatic content [12]. Color regions identified during this phase are approximately homogeneous, but there are several cases in which the segmentation process may produce over-segmented or under-segmented images. This can hinder an effective retrieval due to the difficulty to map regions of similar but not identical images. For each image a graph model is constructed, where each node represents a region and is labelled with a feature vector capturing region area and color. Edges between nodes are used to encode region adjacency.

The example reported, aimed at testing the improvement of retrieval effectiveness determined by the use of the merging strategy applied during graphs comparison. To this end, we compared retrieval results obtained by running the matching algorithm with two different settings of parameter  $\tau_c$  which thresholds the nodes compatibility: in the first case, we used  $\tau_c = 0.3$ , thus allowing the combination of adjacent nodes (*node merging enabled*); in the second case we used  $\tau_c = 0$ , thus preventing all nodes to merge with adjacent nodes (*node merging disabled*). This latter choice reduces the matching problem to the assignment of best fitting nodes in the two graphs.

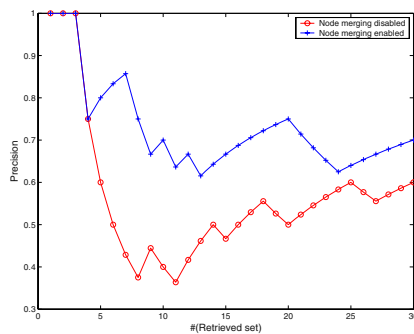
In Fig.3 a manually authored color sketch is used in order to retrieve paintings representing faces. Actually, this is the application scenario which less exploits the potentiality of the proposed approach, since the probability that nodes merging takes place in the query is quite low. This is mainly due to the fact that a user will probably draw a small number of patches identified by very different colors. As a consequence, it is highly probable that node merging will be performed only in the database graphs. So, testing the method in this case should provide a lower bound in the improvement that can be expected in the application of this approach.

The first retrieved images are shown in Fig.3. It can be noticed that the top five retrieved images represent portrait paintings. The last image does not represent any face but is retrieved since its regions have colors similar to the colors used in the query.



**Fig. 3.** Retrieval example based on the matching algorithm. The query image is on the left, followed by the top ranked results.

Fig.4 shows the precision curves obtained by running the matching algorithm with dynamic merging of nodes enabled and disabled, respectively. The horizontal axis represents the size of retrieval set, while the vertical axis is the precision. Values of precision are reported for different sizes of the retrieval set (from 1 to 30), showing that node merging can significantly improve the effectiveness of the retrieval process.



**Fig. 4.** Precision curves obtained by running the matching algorithm with dynamic node merging enabled (upper curve) and disabled (lower curve).

## 5 Conclusions and Future Work

In this paper a novel solution has been proposed for error tolerant graph matching. The solution fits with the class of edit distance based techniques. In particular, the traditional set of edit operations is extended so as to allow node merging during the matching process. An analysis of the computational complexity of the proposed approach has been presented to evidence that the larger the size of the two graphs being compared, the smaller is the increase of complexity associated with management of node merging. Preliminary results are reported to demonstrate the potential and effectiveness of the proposed solution. Future work will address a more extensive experimentation and testing as well as a comparison with alternative techniques for graph matching, both in terms of computational complexity and matching accuracy.

## References

1. H. Bunke. "Recent Developments in Graph Matching". In Proc. 15th Int. Conference on Pattern Recognition (Barcelona, Spain, Sep.3-7), vol. 2, 2000, 117-124.
2. J. Ullman. "An Algorithm for Subgraph Isomorphism". Journal of the ACM. Vol.23, N.1, pp.31-42, 1976.
3. R. Ambauen, S. Fischer, H. Bunke. "Graph Edit Distance with Node Splitting and Merging and its Application to Diatom Identification", In Proc. of Int. Workshop on Graph based Representations in Pattern Recognition, GbRPR-03, LNCS 2726, York, UK, June 30 - July 2, 2003, pp.95-106.
4. H. Bunke. "Error-correcting Graph Isomorphism Using Decision Tree". Int. Journal of Pattern Recognition and Artificial Intelligence, Vol.12, pp.721-742, 1998.
5. A. Massaro, M. Pelillo. "Matching graphs by pivoting". Pattern Recognition Letters, vol. 24, no. 8, pp.1099-1106, 2003.
6. A. Hlaoui, S. Wang. "A New Algorithm for Inexact Graph Matching". In Proc. 16th International Conference on Pattern Recognition (August 11-15, 2002, Quebec City, Canada), Vol.2, pp.465-468, 2002.
7. W.H. Tsai, K.S. Fu. "Error-Correcting Isomorphism of Attributed Relational Graphs for Pattern Analysis". IEEE Trans. on Systems, Man, and Cybernetics, vol.9, n.12, Dec. 1979.
8. B.T. Messmer, H. Bunke. "A new algorithm for error-tolerant subgraph isomorphism detection". IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 20, no. 5, 1998, 493-504.
9. R. Cesar, E. Bengoetxea, I. Bloch. "Inexact Graph Matching Using Stochastic Optimization Techniques for Facial Feature Recognition". In Proc. 16th International Conference on Pattern Recognition (August 11-15, 2002, Quebec City, Canada), Vol.2, pp.465-468, 2002.
10. C. Gomila, F. Meyer. "Tracking Objects by Graph Matching of Image Partition Sequences". In Proc. of Int. Workshop of Graph based Representation for Pattern Recognition (GbRPR 2001), pp.1-11, 2001.
11. L. Gregory, J. Kittler. "Using Graph Search Techniques for Contextual Colour Retrieval". In Proc. of Structural, Syntactic and Statistical Pattern Recognition. LNCS-2396, pp.186-194, 2002.
12. A. Del Bimbo, M. Mugnaini, P. Pala, and F. Turco. "Visual Querying by Color Perceptive Regions", *Pattern Recognition*, vol.31, n.9 pp.1241-1253, 1998.
13. M. Das, E.M. Riseman, B.A. Draper. "FOCUS: Searching for multi-colored objects in a diverse image database". In Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR-97), pp. 756-761, 17-19 June 1997

# STRICT: An Image Retrieval Platform for Queries Based on Regional Content

Jean-Francois Omhover and Marcin Detyniecki

LIP6 – Pole IA  
Universite P. et M. Curie - CNRS  
8 rue du Capitaine Scott  
75015 Paris

**Abstract.** This paper presents a CBIR system that is based on a segmented representation of image content. It compares regional features using fuzzy similarity, which have been shown to be psychologically intuitive. We show that they can be aggregated to support four different type of original queries. The system also supports competitive queries to test different visual comparison measures, and lets the expert user manipulate the parameters of the functions involved.

**Keywords:** Image Retrieval, Segmentation, Fuzzy Similarity Measures, Aggregation.

## 1 Introduction

In the last decade, there has been an increasing interest in image retrieval systems. The idea is to let some user query an image database for a specific image content. Unfortunately, this common goal has not yet been achieved [6].

The community has focused on the automatic extraction and comparison of image features. The query is then expressed not as some textual description, but as an image, chosen by the user. As an answer, the user obtains the list of images in the database that are found most similar to his request. Processing such a comparison is based on a set of signatures extracted from the images and is associated to a similarity measure.

In the last few years, segmentation has been used as a tool for image processing. Images can be segmented in regions, roughly corresponding to objects in the image, thus reflecting the objective content of an image. Measures have been developped to compare regions one to the other.

For the moment, only a few image retrieval systems follow this approach. For instance, Blobworld [1] proposes its segmentation tool to the user, who specifies the request by selecting a set of regions of interest. The retrieval then consists in comparing pairs of regions. The returned images are those which obtained the best scores in the one-to-one region similarity computation. Simplicity [7,2], another system, extends this to image-to-image comparison. It compares images globally by aggregating region-to-region similarities. Unfortunately, its scheme

does not include the spatial configuration of the regions, though it tries to match regions of the request to regions of each image.

In this paper, we present a system called "STRICT". It has been designed to be used as a platform for testing visual similarity measures. It is open to the implementation of new similarity measures. It lets the user modify its parameters online. It is also capable of running several requests in parallel, to compare their effectiveness. It proposes a score-post-processing tool to aggregate the result of different comparison measures. Based on these tools, we propose four original visual requests using the aggregation of global or regional similarity measures.

In the following, we present the architecture of STRICT. In section 3, we introduce the regional features that were implemented. In section 4, we propose four original type of visual requests our system supports. Finally, in section 5, we illustrate our approach with some experimental requests.

## 2 STRICT Image Retrieval Platform

We developed a complete image retrieval system called STRICT [8]. This system has been designed so that it fulfills the five following constraints. First, it proposes image retrieval capabilities based on regions. Secondly, the features of these regions are extracted automatically, in an offline process. Third, as there are many parameters which can be adapted to improve the effectiveness of a visual comparison measure, the system lets the user modify these parameters and instantly run the request using these new values. Four, it also proposes a score post-processing tool to aggregate different measures of similarity. Finally, it runs parallel requests, enabling the user to get the result using different similarity functions, and compare them. In this section, we expose the architecture of STRICT. We also detail the request protocol.

### 2.1 Indexation Server

Our architecture is composed of two distinct systems. The first is our own indexation server that keeps the signature database and computes similarities on demand. The second is an HTTP server that runs the interface, serves the images to the user and presents the results of his requests.

The indexation server remains invisible to the user. It maintains a database of vectors extracted from an images set. These vectors are extracted offline before launching the server. They can be of any type: either global (like an histogram or color moments), regional (the image is first automatically segmented then features are extracted from each region), or even textual (if annotations are given with the image set).

For each indexation process (global, regional, textual), the server maintains the set of features, extracted from each image. Associated to each index, the system keeps a register of similarity measures. Each of these measure is adapted to the index it concerns.

These measures may have parameters. In the register, the server also keeps a list

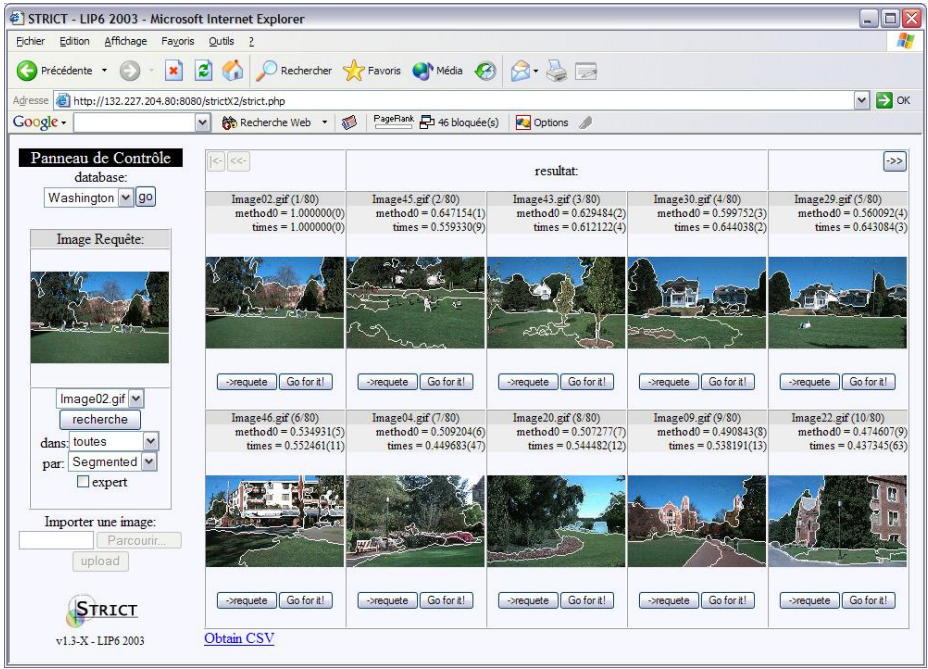
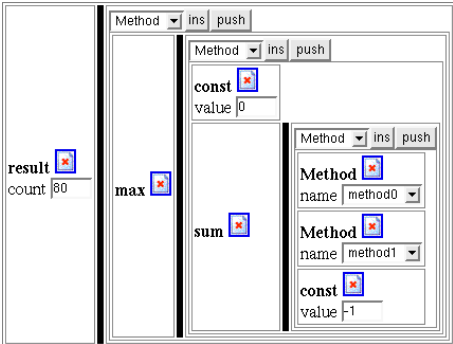


Fig. 1. STRICT Interface, browsing a request's result

of the type, the possible values, and the description of each these paramaters. It also contains a list of mathematical operations that can be used to build a post-processing function.

## 2.2 Web Interface

The indexation server receives the queries from the separate interface. It is a web interface which is the visible part of the system. Its dynamic format enables the expert-user to browse the image database (see figure 1), formulate his queries and see the results using the different similarity measures maintained by the indexation server. When an user opens a new connection to the interface, the description of each measure is sent from the indexation server to the web server. From this list, the web server generates a form for the user that allows to modify the parameters. Thus the interface has not to be modified to cope with modifications or additions of comparison measures or of new parameters. Using the interface, the user can then compose his request with several measures with different parameters. He can choose to run multiple measures. Each measure of the request, when received by the indexation server, is then run against every image of the database. Each score, resulting from each measure specified by the user, is sent back to the interface. When the resulting list of images is displayed, each image is marked by the intensity of each measure specified. This



**Fig. 2.** Lukasiewicz t-norm  $[(x, y) \mapsto \max(x + y - 1, 0)]$  built by simple clicks on STRICT web interface

feature guides the user in the comparison of the effects of its parameters on the retrieval.

The list of the measures involved are not the only information sent in the query. The system can also aggregate these results on-the-fly. The user can formulate an aggregation operator using basic mathematical operations programmed in the indexation server. Using the minimum, the maximum, the addition and the product, he can define his own post-processing score operator (see figure 2). By coupling a list of similarity measures with a post-processing operator, he creates a new measure that he can test against the images. The basic operations programmed in this application let us build most of the known aggregation operators [4]. And it is this feature that let us build interesting visual queries, as shown in section 4.

### 3 Segmented Image Vectors

From an image, a segmentation algorithm extracts regions complying to a given homogeneity criterium. This criterium is usually based on colors: the algorithm tends to isolate regions of connected pixels, which present similar colors. Those regions roughly correspond to the objects present in each image. A similarity measure based on those regions should reflect an objective similarity of the content.

In order to build a semantic similarity between images, we developed a similarity measure for regions based on the similarities of color, shape and position [8]. Here, we briefly present the segmentation algorithm. Then we introduce the definition of fuzzy similarity measures. And finally, we explain which region features we chose to extract and how these features are compared using fuzzy similarities.



### 3.1 Segmentation

For the segmentation of an image in regions, many different approaches exist [3]. They have been developed for thirty years and applied to various application fields. They all aim at building a crisp partition of the image, based on vectors computed from the pixels: color, texture coefficients, edge orientation.

Our algorithm [5] provides good results in a very short computation time. It follows the merge approach: each pixel is first considered as an isolated region, then fusions are operated to merge connected pixels of similar colors, until there is no more possible fusion. This arrives when all the connected regions are dissimilar enough.

### 3.2 Fuzzy Similarity Measures

In [13], Tversky introduces a new definition of similarity measure that breaks with the classical notion of distance. As shown in [11], these measures provide an intuitive measurement of the similarity. They are also independant on the scale of the feature sets. Furthermore, they provide a comparison score that is normalized to  $[0,1]$ , so that it is easy to aggregate. For these three reasons, we implemented them in STRICT to compare the regional features.

Based on a psychological approach, Tversky's work proposes to evaluate the similarity between two sets of binary features by measuring their common and distinctive features. An extension of this definition to fuzzy sets can be used to compare sets of gradual features. Applied to the histograms  $H_A, H_B$  of two images  $A, B$ , this leads to the following computations:

- $M(A \cap B) = \sum_{C_i} \min(H_A(C_i), H_B(C_i))$ , the area of the features that are common to A and B.
- $M(A - B) = \sum_{C_i} \max(H_A(C_i) - H_B(C_i), 0)$ , the area of the features that distinguish A from B.
- $M(B - A) = \sum_{C_i} \max(H_B(C_i) - H_A(C_i), 0)$ , the area of the features that distinguish B from A.

According to Tversky, the similarity between  $A$  and  $B$  is a functions as these three areas. It is to notice that the classical histogram intersection proposed by Swain and Ballard [12] and all the successors are particular cases of this framework. Here, we chose to implement three other: Jaccard, Dice and Ochiai [10]. With  $X = M(A \cap B)$ ,  $Y = M(A - B)$ ,  $Z = M(B - A)$ , we have:

$$\begin{aligned} S_{jaccard}(X, Y, Z) &= \frac{X}{X+Y+Z} \\ S_{dice}(X, Y, Z) &= \frac{2X}{2X+Y+Z} \\ S_{ochiai}(X, Y, Z) &= \frac{X}{\sqrt{(X+Y)}\sqrt{(X+Z)}} \end{aligned}$$

The fact of considering different measures enable us to discover interesting properties. In particular, we show in [9] that the result ranking list provided by an information retrieval system is conserved in several cases.

### 3.3 Region Color, Shape, and Position

The resemblance between regions is more elaborated than the global one. We have to consider several aspects of comparison, which are color, shape and position. For each of these comparisons, a region is represented as a vector. Because these three vectors belong to different spaces, and cannot be related one to the other, the similarity between pairs of region relies on three different measures: one measure for each of the vector pairs. For two regions  $R(i)$  and  $I(j)$ , extracted respectively from the image request  $R$  and from an image database entry  $I$ , the region similarity  $S_{reg}$  can be written as a weighted mean of these three "sub-measures". With  $\lambda_c + \lambda_s + \lambda_p = 1$  :

$$\begin{aligned} S_{reg}(R(i), I(j)) = & \lambda_c \cdot S_{reg|color}(R(i), I(j)) \\ & + \lambda_p \cdot D_{reg|position}(R(i), I(j)) \\ & + \lambda_s \cdot S_{reg|shape}(R(i), I(j)) \end{aligned}$$

In our experiments, we have set the respective weights of these three measures to  $\lambda_c = 0.6$ ,  $\lambda_s = 0.2$ ,  $\lambda_p = 0.2$ .

*Fuzzy Region Color Similarity* The region color similarity  $S_{reg|color}$  is based on the color histograms of the regions. To compare the regional color histograms, we use measures presented in section 3.2.

*Proximity Measure* The proximity measure  $D_{reg|position}$  is the fuzzy inverse ( $x \rightarrow 1 - x$ ) of the normalized distance between the geometric centers of two regions. This distance is normalized so that the distance between the opposite corners of the images equals 1.

*Fuzzy Shape Similarity* The shape similarity measure  $S_{reg|shape}$  is based on the centered binary masks of the regions. A mask is the matrix  $K_{R(i)}$  of  $\{0, 1\}$  that represents the crisp membership of each pixel of the image  $R$  to the region  $R(i)$ .  $K_{R(i)}$  is centered so that its central point gives the membership of the geometric center of  $R(i)$ .

To compare the two shapes of regions  $R(i), I(j)$ , we compute their common and distinctive pixels, and apply the similarity measures of section 3.2.

## 4 Interesting Query Examples

STRICT can compute global and regional similarities. For the moment, the system is based on one-to-one comparison functions. For a single request image  $R$ , a similarity measure  $S$  is run against every entry  $I$  of our image database. The value  $S(R, I)$  of the comparison is computed and returned for post-processing. We adapt the global image approach to regions obtained using an automatic segmentation of the image. A regional similarity measure  $S_{reg}$  is then used to find in every image  $I$  a region that is similar to a single region  $R(i)$  of image  $R$ . The returned score is noted  $S_{reg}(R(i), I)$  and it corresponds to the truth value of the proposition "there exists a region like  $R(i)$  in  $I$ ".

Those scores, corresponding to the similarity, can be aggregated on-the-fly under users' command. An operator, noted  $Agg$ , is used to procure a single value  $f(I)$

from the  $n$  different similarities involved in the query. This feature enables the user to build different kinds of requests. This section points out four interesting schemes, where aggregation is the key element.

*Multiple features:* Using a single image request  $R$ , the user can compare the images on the basis of multiple features (color, texture...). STRICT then aggregates the scores resulting from the different similarity measures. The final score  $f(I)$  of the image  $I$  is then expressed as:

$$f(I) = \text{Agg}(S_1(R, I), \dots, S_n(R, I))$$

where  $\text{Agg}$  can be the averaging operator, the minimum, the maximum, or any buildable operator (see section 2). For instance, the user can use a weighted mean to give more importance to the color against the texture.

*Multiple image requests:* The user can also specify multiple image requests. The idea is to retrieve images, which are similar to a set of examples (some of the examples might be more important than others). Based on  $R_1, \dots, R_n$ , a set of request images, STRICT will compute the global similarity between each image  $I$  in the database and each  $R_i$ . It will then aggregate these similarity to render a final score  $f(I)$  for each  $I$ .

$$f(I) = \text{Agg}(S(R_1, I), \dots, S(R_n, I))$$

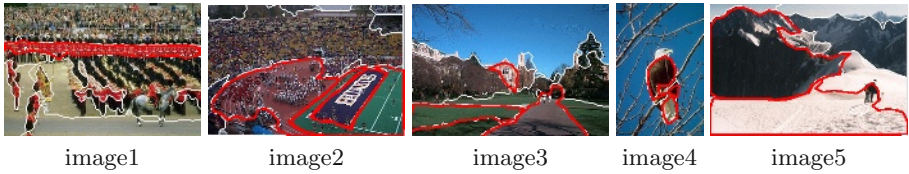
The aggregation used must correspond to the goal of the request. For instance, setting  $\text{Agg}$  as the maximum will retrieve the images that are similar to *at least one* of the images  $R_1, \dots, R_n$ . Instead, setting  $\text{Agg}$  as the minimum will retrieve images that are similar to *all* of the request images. In other words, the retrieved image has to be similar to each of the images of the request. Using the average will let the user weight some of the  $R_i$ , and therefore retrieve images that look like the strongly weighted images. In this last case, the user can also use a fuzzy negation applied to a similarity measure  $S(R_i, I)$  to express that he wants images that does not look like  $R_i$ , and that are still similar to the other inputs.

*Multiple regions:* On a single image request  $R$ , the user can select different regions  $R(i), \dots, R(j)$  that he wants to look for in one image of the database. The final score  $f(I)$  is then expressed by:

$$f(I) = \text{Agg}(S_{\text{reg.}}(R(i), I), \dots, S_{\text{reg.}}(R(j), I))$$

Attention must be paid to the fact that each score  $S_{\text{reg.}}(R(i), I)$  is an aggregated one (see 3): the score does not indicate if the set of regions in the image  $I$  is formed of the same number of regions, they just individually look like one of the regions  $R(i)$ .

As in the global case, the choice of  $\text{Agg}$  will change the request into a specific purpose. By setting  $\text{Agg}$  as the maximum, the user will get images containing at least one of the specified regions. With the minimum, he will get the images containing all of them. Again, he can assign weights by using an averaging operator. He can also specify regions he don't want to see in the retrieved images.



**Fig. 3.** Images used as requests to evaluate the efficiency of region similarity

*Regions from different images:* Obviously, the previous scheme can also be applied on a request composed of *regions extracted from different images*.  $f(I)$  then takes the following form:

$$f(I) = \text{Agg}(S_{reg.}(R_1(i_1), I), \dots, S_{reg.}(R_n(i_n), I))$$

In this case, the user compose a request from different sources. It is like he would build a squetch from different example regions.

The dynamic interface of our system is a practical tool to test all of these four schemes. With simple actions, an expert user can simulate one of these four requests, specify its parameters, and compose an aggregator to fulfill its needs. No current CBIR system offers all of these features. Though the similarity measures in our system are not fully developped yet, we have build a structure that let us implement, experiment and compare different methods of the image retrieval field.

## 5 Experimental Queries

As our system has been recently developed, we have not proceeded yet to its full evaluation. In this section, we only propose a preliminary essay based on five visual requests (see figure 3): each time, 2 or 3 regions of interest were selected, then used as a query. We choose to aggregate regional similarity measures (based on color, position and shape) using an averaging operator. The effectiveness of this aggregated measure is compared to the effectiveness of the global histogram similarity (one global histogram is extracted and compared to another histogram by a fuzzy similarity measure). The system retrieves the 80 best results for both methods among a database of 7700 images. This operation takes about 1.2 seconds. We evaluate the effectiveness of both methods by measuring two classical values:

- the rank RK of the first non pertinent image in the list.
- the proportion P of pertinent images in the first 10 results (extended to 20 or 30 when the first non-pertinent image was not present in the 10 first). The number of 10 results was chosen only to reflect the satisfaction of the first page of the results returned.

For a method to be efficient, it has to maximise the rank of the first non pertinent image, and also maximise the proportion of pertinent images in the first results.

On table 1, we observe that our method based on region matching gets better results than the global histogram comparison. Though the present preliminary evaluation shows encouraging results, a larger benchmarking should be runned to prove the effectiveness of our system.

**Table 1.** Evaluation Results

	Segmented		Global Histo.	
	P	RK	P	RK
image1	19/20	19	7/20	5
image2	22/30	19	24/30	21
image3	4/10	5	4/10	3
image4	8/10	6	4/10	2
image5	4/10	4	4/10	2
average results				
	0.65	10.6	0.47	6.6

## 6 Conclusion

Our image retrieval system uses a regional representation of images, built using a segmentation algorithm. This solution, currently investigated in the community, has been implemented in STRICT to let the user specify his visual interest in the request. For the expert, our system proposes features that enables him to modify parameters, build complex requests, and compare their results in a dynamic and easy to use interface.

We realize that the choice of the parameters and the operators used by the system to build complex queries may be difficult for the non-expert users. STRICT is in fact a CBIR experimentation platform that requires from users to know what they are doing, and for what purpose. A simpler version of the interface system is being developed to propose a ready-to-use image retrieval tool, including pre-chosen parameters and aggregation operators.

## References

1. C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In *Third Int. Conf. on Visual Information Systems*, Amsterdam, 1999.
2. Y. Chen and J.Z. Wang. A region-based fuzzy feature matching approach to content-based image indexing and retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(9), September 2002.
3. H.D. Cheng, X.H. Jiang, Y. Sun, and J. Wang. Color image segmentation: advances and prospects. *Pattern Recognition*, 34:2259–2281, 2001.
4. M. Detyniecki. *Mathematical Aggregation Operators and their Application to Video Querying*. PhD thesis, Universite Pierre et Marie Curie, 2000.

5. G. Durand and P. Faudemay. A fast region-merging segmentation algorithm for video analysis and indexing. In *CIR 98 Symposion and Workshop on Image Retrieval*, Newcastle, 1998.
6. J.P. Eakins. Towards intelligent image retrieval. *Pattern Recognition*, 35:3–14, 2002.
7. J. Li, J.Z. Wang, and G. Wiederhold. Irm: Integrated region matching for image retrieval. In *8th ACM Int. Conf. on Multimedia*, pages 147–156, 2000.
8. J.F. Omhover, M. Detyniecki, and B. Bouchon-Meunier. A region-similarity-based image retrieval system. In *IPMU'04*, Perugia, Italy, 2004 (submitted).
9. J.F. Omhover, M. Detyniecki, and M. Rifqi. Image retrieval using fuzzy similarity : Measure equivalence based on invariance in ranking. In *IEEE International Conference on Fuzzy Systems Fuzz-IEEE*, Budapest, Hungary, July 2004.
10. M. Rifqi, M. Detyniecki, and B. Bouchon-Meunier. Discrimination power of measures of resemblance. In *IFSA'03*, Istanbul, Turkey, 2003.
11. S. Santini and R. Jain. Similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):871–883, 1999.
12. M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.
13. A. Tversky. Features of similarity. *Psychological Review*, 84:327–352, 1977.

# Improved Video Content Indexing by Multiple Latent Semantic Analysis

Fabrice Souvannavong, Bernard Merialdo, and Benoît Huet\*

Département Communications Multimédias

Institut Eurécom

2229, route des crêtes

06904 Sophia-Antipolis - France

(souvanna, merialdo, huet)@eurecom.fr

**Abstract.** Low-level features are now becoming insufficient to build efficient content-based retrieval systems. Users are not interested any longer in retrieving visually similar content, but they expect retrieval systems to also find documents with similar semantic content. Bridging the gap between low-level features and semantic content is a challenging task necessary for future retrieval systems. Latent Semantic Analysis (LSA) was successfully introduced to efficiently index text documents by detecting synonyms and the polysemy of words. We have successfully proposed an adaptation of LSA to model video content for object retrieval and semantic content estimation. Following this idea we now present a new model composed of multiple LSA's (M-LSA) to better represent the video content. In the experimental section, we make a comparison of LSA and M-LSA on two problems, namely object retrieval and semantic content estimation.

## 1 Introduction

Because of the growth of numerical storage facilities, many documents are now archived in huge databases or extensively shared over the Internet. The advantage of such mass storage is undeniable. However the challenging tasks of multimedia content indexing and retrieval remain unsolved without the expensive human intervention to archive and annotate contents. Many researchers are currently investigating methods to automatically analyze, organize, index and retrieve video information [1,2]. This effort is further stressed by the emerging MPEG-7 standard that provides a rich and common description tool of multimedia contents. It is also encouraged by Video-TREC<sup>1</sup> which aims at developing video content analysis and retrieval.

One of the major task is to bridge the gap between low-level features and the semantic content. To address this problem we propose a new robust method to index video shots. Based on our previous work on Latent Semantic Analysis (LSA) for object retrieval in [3]

---

\* This research was supported by the EU project GMF4iTV under the IST-programme (IST-2001-34861)

<sup>1</sup> Text REtrieval Conference. Its purpose is to support research within the information retrieval community by providing the infrastructure necessary for large-scale evaluation.  
<http://trec.nist.gov>

and semantic content estimation in [4], we present a new model that uses Multiple Latent Semantic Analysis. LSA has been proven effective for text document analysis, indexing and retrieval [5]. The key idea is to map high dimensional count vectors to a lower dimensional space so-called latent semantic space. Some extensions to audio and image features were then proposed [6,7]. In 1999, a probabilistic framework, called PLSA, was introduced for text document indexing in [8]. Then authors of [9] have recently made a comparison of both methods, i.e. LSA and PLSA, for image auto-annotation. They conclude that classic LSA model defined on a very basic image representation performs as well as much more complex state-of-the-art methods and outperforms PLSA. In this paper, we propose a new method that relies on Multiple Latent Semantic Analysis. The underlying idea is to group shots in order to better detect the latent semantic that locally resides in groups and that might be covered by a global approach.

The first part briefly presents the adaptation of LSA to video content modeling. Next we present Multiple Latent Semantic Indexing. Then experimental results are presented and commented to finish with the conclusion and future work.

## 2 Latent Semantic Analysis

Latent Semantic Analysis (LSA) has been proven efficient for text document analysis and indexing. Contrary to early information retrieval approaches that used exact keyword matching techniques, it relies on the automatic discovery of synonyms and the polysemy of words to identify similar documents. We proposed in [3] an adaptation of LSA to model the visual content of a video sequence for object retrieval. We summarize the proposed solution in the following before presenting our new approach.

Let  $V = \{S_i\}_{1 \leq i \leq N}$  be a sequence of shots representing the video. Usually many shots contain the same information but expressed with some inherent visual changes and noise. Latent Semantic Analysis is a solution to remove the noise and find equivalences of the visual content to improve shot matching. It relies on the occurrence information of some features in different situations to discover synonyms and the polysemy of features. A classical approach is to use the singular value decomposition (SVD) of the occurrence matrix of features in shots to achieve this task. The content of shot  $i$  is described by a raw feature vector  $r_i$ , such as color histogram, gabor's energies, motion, ... However such feature vectors suffer from the loss of spatial information by keeping only global features. To overcome this problem, a more appropriate signature is used. First of all, frames composing shots are segmented into homogeneous regions. Similar regions, described by raw feature vectors, are then clustered in groups where they are finally mapped. The representative of each cluster is then called a visual term while the set of clusters is called the dictionary. Shots are now represented by the count of visual terms that describes the content of their regions. Let now denote  $q$  this new feature vector. The singular value decomposition of the occurrence matrix  $C$  of visual terms in shots gives:

$$C = UDV^t \quad \text{where} \quad U^t U = V^t V = I \quad (1)$$

With some simple linear algebra we can show that a shot (with a feature vector  $q$ ) is indexed by  $p$  such that:

$$p = U^t q \quad (2)$$



$U^t$  is then the transformation matrix to the latent space. The SVD allows to discover the latent semantic by keeping only the  $L$  highest singular values of the matrix  $D$  and the corresponding left and right singular vectors of  $U$  and  $V$ . Thus,

$$\hat{C} = U_L D_L C_L^t \quad \text{and} \quad p = U_L^t q \quad (3)$$

The latent space of size  $L$  is now ready for improved shot comparison thanks to the cosine measure. The number of singular values kept drives the LSA performance. On one hand if too many factors are kept, the noise will remain and the detection of synonyms and the polysemy of visual terms will fail. On the other hand if too few factors are kept, important information will be lost degrading performances. Unfortunately no solution has yet been found and only experiments allows to find the appropriate factor number.

In [4], we also noticed that the creation of a visual dictionary has a major disadvantage when dealing with many videos: it introduces differences between regions, i.e. due to the mapping, that might be too important. To diminish this side effect of the mapping, regions are mapped to their  $k$ -closest visual items and this conducted to a better performance.

### 3 Multiple Latent Semantic Analysis

Visual content carries an extremely rich information and LSA through SVD is a simple linear approach to model this diversity. We propose to introduce Multiple Latent Semantic Analysis (M-LSA) to describe the content and *locally* find its “latent semantic”. M-LSA involves two steps. Firstly we find  $K$  homogeneous partitions  $P_k$  in the feature space with respect to training shots. We then apply classical LSA to model each area and find  $K$  latent spaces. Secondly we index shots with respect to this decomposition and modeling of the feature space. We now thoroughly describe the method.

#### 3.1 Local Latent Semantic Analysis

In order to improve the effect of SVD which is a linear transformation, we propose to apply the SVD locally in homogeneous partitions  $P_k$  of the feature space. This operation aims at detecting singular directions more accurately in local areas of the feature space. Thus we expect the LSA to be locally more efficient. Training shots allow to construct an efficient partition with respect to the content and one way to proceed is to use  $k$ -means algorithm on training shots.

Once the feature space is partitioned, we construct matrices  $C_k$  that contain the occurrences of visual terms in shots belonging to the partition  $k$ . We then apply classical LSA to model each area. Thus,

$$C_k = U_k D_k V_k^t \quad (4)$$

In this situation where  $k$  models are constructed, it is difficult to select the appropriate number of factors  $L_k$  kept per model. Empirically, we select a single value  $l$ , the selection coefficient, that gives the percentage of factor involved in each model to make the projection. Finally,

$$\hat{C}_k = U_{L,k} D_{L,k} V_{L,k}^t \quad (5)$$

where  $L = l \times \min(\text{number of shots, number of visual terms})$

In the following,  $U_{L,k}$  is denoted  $U_k(l)$  for convenience.

### 3.2 Indexing with Local LSA

From the presented decomposition and modeling of the feature space, we derive a new representation of shots. A direct approach is to index shots with respect to the partition where they are located. A shot signature is then composed of a partition number and its projection in the associated left singular space. Let  $q \in P_k$  and  $p = U_k^t(l)q$

$$\text{sim}(q, q') = \begin{cases} \cos(p, p') & \text{if } q' \in P_k \quad (p' = U_k^t(l)q') \\ -1 & \text{else} \end{cases} \quad (6)$$

Unfortunately, shots can not be compared between partitions but only intra-partition comparisons are possible. This drawback becomes particularly important when looking for an object, i.e. only a subpart of the shot. In that case the query is not well classified in partitions that are homogeneous only with respect to complete shots. This suggests to project shots in all partitions, compare shots in each partition and then combine similarity measures to form a single-valued similarity measure. This second approach suffers from the fact that the projection errors can be high whereas the projected shots are close. In this case even if projected shots are similar, we can not guarantee that shots are similar. To take into account all these parameters, we derive a similarity measure of the form:

$$\text{sim}(q, q') = \max_i \{ (\cos(q, \hat{q}_i) \cos(q', \hat{q}'_i))^2 \cos(p_i, p'_i) \} \quad (7)$$

$$\begin{aligned} \text{where } p_i &= U_i^t(l)q \quad \text{and} \quad \hat{q}_i = U_i(l)p \\ \text{and } p'_i &= U_i^t(l)q' \quad \text{and} \quad \hat{q}'_i = U_i(l)p' \end{aligned}$$

The first two cosine functions measure the similarity between shots and their reconstruction when using the local model  $i$ . Cosine functions are used to obtain normalized values whatever values are taken by the selection coefficient. We then take the square of the product to diminish the impact of projection errors. The third cosine function measure the similarity between projected shots with the local model  $i$ . This similarity measure has the property to favor similar shots in a partition where they are both well projected. However, indexed shots need the value  $\cos(q, \hat{q}_i)$  and the vector  $p_i$ , thus the selection coefficient can not be easily changed without computing again the value of the cosine. In future work we are going to evaluate a measure of the form:

$$\text{sim}(q, q') = \max_i \frac{\cos(p_i, p'_i)}{\frac{\|p_i^\perp\|}{\|p_i^\perp + p_i\|} \frac{\|p'^{\perp}_i\|}{\|p'^{\perp}_i + p'_i\|}} \quad (8)$$

## 4 Experiments

This new Multi Latent Semantic Analysis approach is evaluated on two different tasks. First the system performance is measured in the framework of object retrieval on a short set of cartoons (approximately 10 minutes) from the MPEG-7 data set. Then, its is evaluated in the context of Video-TREC feature extraction.

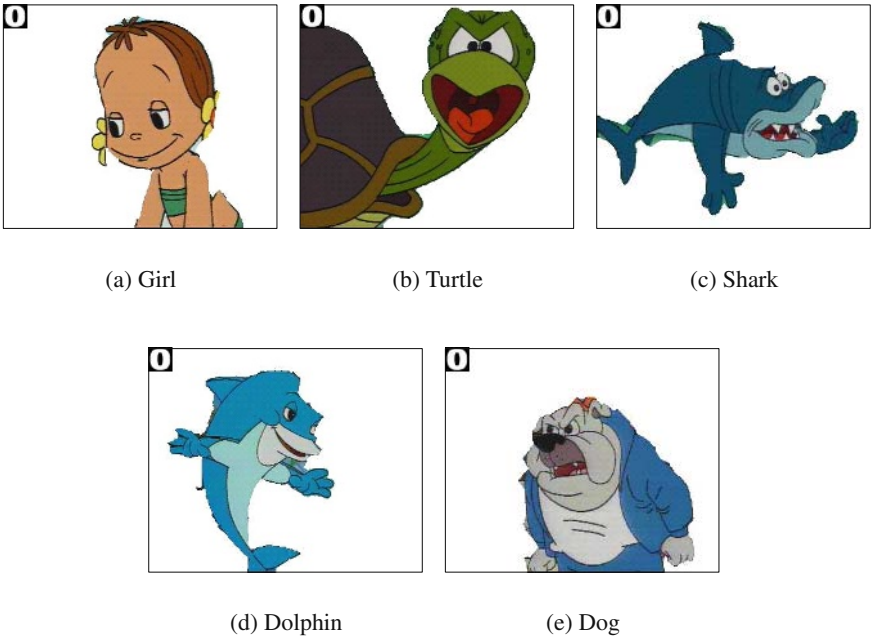
## 4.1 Object Retrieval

The object retrieval evaluation is conducted on Docon's production donation to the MPEG-7 dataset. First the video sequence is subsampled by keeping one frame per second. Selected frames are then segmented into regions ([10]) described by a 32 bins HS histogram. To measure the performance a ground truth has been established and 5 different objects were selected and annotated in 950 frames, see figure (4.1) for an illustration. 17 to 108 queries are possible per object with a total of 245 queries. The mean precision is computed to have a global overview per object in figure (2(a)). The partition size is 2, i.e. two local LSA's. And the curves are the result of extensive experiments conducted to select the best number of factors for each model. A selection coefficient of 5.2% was kept for LSA method, yielding to a latent space of 39 features. A selection coefficient of 4% was kept for M-LSA method, yielding to two latent spaces of size 17 and 21. Thus, we have indexing signatures of reasonable and similar size in both cases. The first plot reveals the interest of M-LSA which outperforms LSA on shark, dolphin and dog objects. Figure (2(b)) shows the evolution of the mean precision over all possible queries with respect to standard recall values. M-LSA improves the stability of the IR system. However performances are under our expectation. This might be due to the video length that is too short. Indeed the dictionary computed from all regions with the k-means clustering has a size of 750. There are less shots than visual terms in partitions. Latent spaces have not enough samples to correctly remove noise and discover synonyms and we expect more improvements when enough data are available to train transformations to local latent spaces.

## 4.2 Video-TREC Feature Extraction

Our system is also evaluated in the context of Video-TREC. One task is to detect the semantic content of video shots. 17 features were proposed: (1) Outdoors, (2) News-subject, (3) People, (4) Building, (5) Road, (6) Vegetation, (7) Animal, (8) Female-speech, (9) Car-truck-bus, (10) Aircraft, (11) News-subject-monologue, (12) Non studio-settings, (13) Sporting-event, (14) Weather, (15) Zoom-in, (16) Physical-violence and (17) Madeleine Albright. For each feature, 30.000 test shots are ordered with respect to their detection score value. Then the average precision at 2,000 shots is computed to characterize the performance of the system for each feature. We have proposed in [4] a simple approach using k-nearest neighbors on LSA features to estimate shot semantic features and compute their detection score. The "training" set is constituted of 44.000 shots and 17.000 were used to build the latent space. For this difficult task, two dictionaries are used: one containing color terms through 32 bins HS histograms and the other containing texture terms through 24 gabor's energies. Shots are reduced to their key-frame to save computation efforts. Indeed the segmentation process is very time consuming and untractable for all frames of the database. Visual terms are most representatives regions present in all key-frames and the signature of a shot is simply the count of visual terms where its key-frame regions are mapped. Similarity measures are independently computed for each feature type and then combined as follows:

$$sim(q, q') = w_c \times sim_{color}(q, q') + w_t \times sim_{texture}(q, q') \quad (9)$$

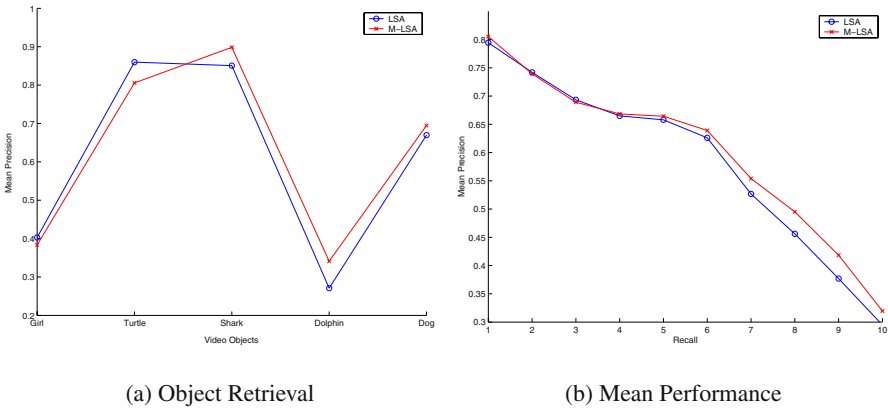


**Fig. 1.** An illustration of the five selected and annotated objects in Docon’s production cartoons.

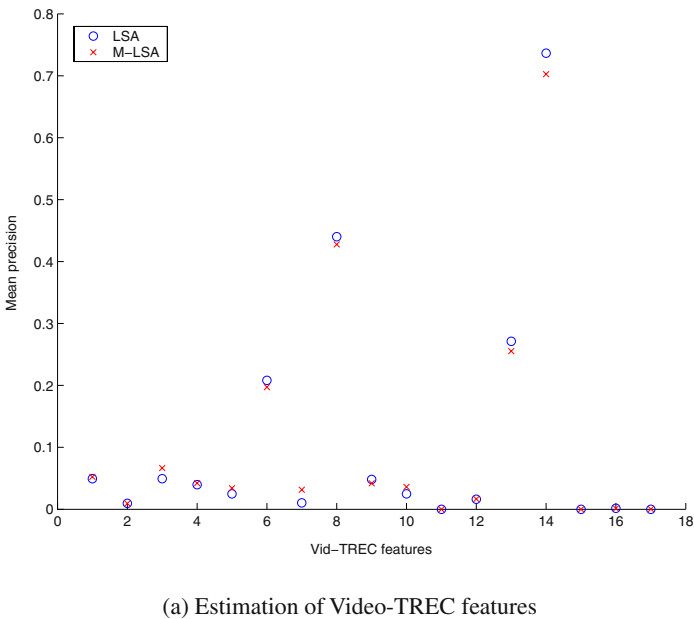
For simplicity  $w_c = w_t = 1$  knowing that the appropriate selection of weights can be included in a training algorithm. The figure (3(a)) compares performances of the proposed M-LSA and LSA approaches. Given the volume of data to process, the tuning of parameters is very time consuming. Thus the selection coefficient  $l$  is empirically set to 10%.

**5 Conclusion and Future Work**

Our previous work on Latent Semantic Analysis revealed the high potential of this simple method for object retrieval and semantic content estimation. In this paper we have presented a new approach to model video content with Latent Semantic Analysis. In particular we introduced multiple latent spaces to better represent the content. The feature space defined by video shots is decomposed into partitions where LSA’s models are defined. This new representation of the content in multiple latent spaces rises the problem of indexing. We have proposed a method to index and compare video shots in this framework by taking into account shot similarities and projection errors. The method is then evaluated on object retrieval and semantic content estimation problems. A slight improvement is observed for the task of object retrieval, but results are more lukewarm when dealing with semantic content estimation.



**Fig. 2.** Object retrieval performance evaluation. The first figure shows individual performances of the system for each object, while the second curve is the mean precision curve for standard recall values.



**Fig. 3.** M-LSA compared to LSA for the difficult problem of semantic content analysis.

Future work will concern the study of methods to improve the effectiveness of the similarity measure and to select factors in a more appropriate way. We are also interested in looking to probabilistic approaches to build mixture of models that is a very interesting extension to the proposed method. On the other hand, efforts will be provided to construct

more sophisticated shot signatures and include more raw features such that motion, audio and text.

## References

1. Chang, S.F., Chen, W., Meng, H., Sundaram, H., Zhong, D.: A fully automated content-based video search engine supporting spatiotemporal queries. In: *IEEE Transactions on Circuits and Systems for Video Technology*. Volume 8. (1998) 602–615
2. Naphade, M., Kristjansson, T., Frey, B., Huang, T.: Probabilistic multimedia objects (multi-jects): a novel approach to video indexing and retrieval. In: *IEEE International Conference on Image Processing*. Volume 3. (1998) 536–540
3. Souvannavong, F., Merialdo, B., Huet, B.: Video content modeling with latent semantic analysis. In: *Third International Workshop on Content-Based Multimedia Indexing*. (2003)
4. Souvannavong, F., Merialdo, B., Huet, B.: Latent semantic indexing for video content modeling and analysis. In: *The 12th Text REtrieval Conference (TREC)*. (2003)
5. Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., Harshman, R.A.: Indexing by latent semantic analysis. *Journal of the American Society of Information Science* **41** (1990) 391–407
6. Kurimo, M.: Indexing audio documents by using latent semantic analysis and som. In Oja, E., Kaski, S., eds.: *Kohonen Maps*. Elsevier (1999) 363–374
7. Zhao, R., Grosky, W.I.: From features to semantics: Some preliminary results. In: *International Conference on Multimedia and Expo*. (2000)
8. Hofmann, T.: Probabilistic latent semantic indexing. In: *ACM SIGIR*. (1999)
9. Monay, F., Gatica-Perez, D.: On image auto-annotation with latent space models. In: *ACM Multimedia*. (2003) 275–278
10. Felzenszwalb, P., Huttenlocher, D.: Efficiently computing a good segmentation. In: *IEEE Conference on Computer Vision and Pattern Recognition*. (1998) 98–104

# Three Interfaces for Content-Based Access to Image Collections

Daniel Heesch and Stefan Rüger

Department of Computing, Imperial College  
180 Queen's Gate, London SW7 2BZ, England  
{daniel.heesch,s.rueger}@imperial.ac.uk

**Abstract.** This paper describes interfaces for a suite of three recently developed techniques to facilitate content-based access to large image and video repositories. Two of these techniques involve content-based retrieval while the third technique is centered around a new browsing structure and forms a useful complement to the traditional query-by-example paradigm. Each technique is associated with its own user interface and allows for a different set of user interactions. The user can move between interfaces whilst executing a particular search and thus may combine the particular strengths of the different techniques. We illustrate each of the techniques using topics from the TRECVID 2003 contest.

## 1 Introduction

Being able to endow systems with the capacity to exhibit intelligent, human-like behaviour has been an early hope of computer scientists. The past fifty years have seen the gradual erosion of this hope and the consensus seems reached that the early optimism of this research program was ill-founded. The general vision problem, that is the problem of being able to describe the content of a visual scene, is among those problems that have as yet been left untouched by the otherwise relentless progress in computer science. To solve it, we will have to come up with answers to deep and fundamental questions about representation and computation that lie at the very core of human intelligence. This is what renders the problem of content-based image retrieval (CBIR) very exciting and challenging at the same time. The increasing interest in human-computer interaction is testimony of a growing awareness that humans are currently still the most intelligent part of the system and that a tighter integration between humans and machines can lead to results that would otherwise remain unattainable [8]. Unlike in typical computer vision applications, content-based image retrieval (CBIR) systems have an end user seeking information, and thus a dialogue between user and machine seems more adequate from the outset. The presence of a user adds to the problem of image understanding the problem of user understanding, a problem that can evidently only be resolved by incorporating the user in the retrieval process.

We introduce two techniques for content-based image retrieval, and one technique for content-based image browsing. The first technique uses relevance feedback applied to the retrieval results to update weights of the similarity function. The technique comes with an interface that allows users to give continuous feedback. With the second technique we introduce a novel idea that elegantly bypasses the problem of initial feature weighting by gathering the top-ranked images from a multitude of retrieval processes, each carried out with a different weight set. The resulting set of images, which we call the  $NN^k$  of the query ( $NN$  for nearest neighbour, and  $k$  for the number of features), can each be associated with a weight set that, when chosen, will retrieve similar images. It is thus a two-step process that engages the user in relevance feedback after the first step. The third technique also relies on the  $NN^k$  idea, but instead of determining the  $NN^k$  for a query at run-time, the  $NN^k$  of each image from within a collection are determined beforehand and internal links established between each image and its  $NN^k$ . The resulting network provides our basis for content-based image browsing. Details of and evaluative studies on each of the three techniques have been presented elsewhere (e.g. [5], [7], [4]). This paper will place greater emphasis on interface design. We illustrate the functionality of the system by looking in detail how a real search task could be executed using the test collection of TRECVID 2003 and particular search topics thereof. The key contribution of this paper is the development of an integrated interface that combines the strengths of three recently developed techniques for CBIR.

The paper is structured as follows. In Section 2, we briefly discuss work that is related to ours. Section 3 establishes methodological commonalities of the various techniques presented here and includes a brief description of the collection used, the image representations and the method of similarity computation. Section 4-6 introduces the three techniques along with the associated visualizations and user interactions. We summarize and conclude our paper in Section 7.

## 2 Related Work

Relevance feedback as a particular form of human-machine interaction has become a core paradigm in information retrieval and in particular so in CBIR, where it has been shown to substantially improve retrieval performance (e.g. [9], [10]). Few systems address the problem of how to intelligently weigh features prior to the first retrieval, although it is clear that the efficacy of relevance feedback hinges on a satisfactory first retrieval result as little can be learnt from negative examples alone [5]. Aggarwal et al. [1] have presented an interesting two-step approach to feature weighting. The first step involves modifying the query representation by moving it along each of the feature dimensions, and to regenerate from the thus altered representations a set of query images on which feedback is then given. The second step is the retrieval step itself using the newly learnt weights. The technique appears to improve performance but places constraints on the set of features that can be used. It is methodologically different from but in spirit very similar to our idea of  $NN^k$  search.



Visualization of search results has initially taken the form of a 2-d grid layout (e.g. [3]). More recent visualizations aim to preserve the distances of the retrieved images to the query ([5], [13]) and the distances between returned images as in [12]. One of the problems of plane-filling visualizations is the potential overlap between images which we avoid in one of the search result visualizations and minimize in the other.

The importance of browsing as a method of accessing image collections has increasingly been recognized in recent years (e.g. [2], [11]). In the ostensive browsing model developed in [2], the user browses along a dynamically generated tree. The set of possible branches a user may take from a given image depends no less on that image than on the history of past images. The tree is generated dynamically and the method designed to deal with changing information needs. In the model presented by Santini et al. [11], the user can effect a transformation of the image space upon relocating images on the screen. The user can explore the vicinity of a particular region but the scope for fast navigation through the image space is limited. The browsing structure on which our third technique relies on seeks to allow for both the exploration of an image's surrounding as well as efficient browsing, thus allowing for target as well as undirected search.

### 3 Collection, Features, and Similarity Computation

We illustrate the visualizations using the test collection of TRECVID 2003, a collection that contains more than 32,000 key frames from news video sequences. We implemented eleven low-level texture and colour features and made use of the text from the speech recognition transcripts supplied by Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur. For details of these low-level features see [4].

The overall similarity between two images  $X$  and  $Y$  is given by the weighted sum of the feature-specific similarities, i.e.  $S(X, Y) = \sum_i w_i s_i(X, Y)$  where the weights are constrained to sum to one with  $w_i \in [0, 1]$ , and  $s_i$  are feature-specific similarity functions (the  $l_1$  norm for all features). Because the overall similarity is computed as the weighted sum of these feature-specific similarities, we normalize the feature-specific similarities before aggregation such that their medians lie around 1.

## 4 Search with Relevance Feedback

### 4.1 Relevance Feedback Technique

Unlike most systems employing relevance feedback, our technique allows continuous feedback to be given. Given a set of system-computed similarities, the user provides a set of new similarities along a continuous range by relocating images on the screen. We then compute a new set of weights by minimizing the sum of squared errors between the two sets of similarities as described more fully in [5]. The system uses the updated weights for the next retrieval.

## 4.2 Visualization of Search Results and Relevance Feedback Interaction

Search results are displayed in the form of a spiral with an image's distance from the center being proportional to the distances computed by the system. A similar technique has subsequently been used in [13] where the spiral is constrained to be Archimedean, and the distance of an image to the query is indicated by the distance of the image to the center of the screen along the arc of the spiral. Relevance feedback is given on the displayed images by moving them closer towards or further away from the center. Figure 1 shows the retrieval results with an initially equal weight set and the result following relevance feedback on positive examples (pitcher from behind throwing a ball towards the batter).

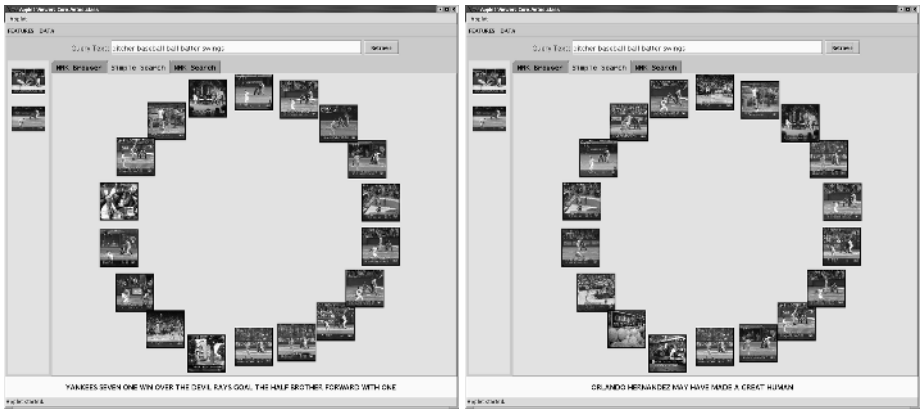


Fig. 1. Result before and after relevance feedback using the Baseball topic

Because the initial result set is already quite satisfactory, extensive feedback can be given, leading to a further increase in the number of relevant images. The technique is useful when the search task is relatively easy or once the weights have been brought into the vicinity of the weight optimum using, for example, the  $NN^k$  technique to be described in the next section.

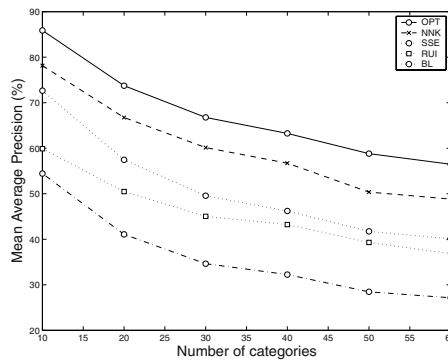
## 5 $NN^k$ Search

### 5.1 A Two-Step Technique for Feature Weighting

The  $NN^k$  idea has first been introduced and explained in greater detail in [6]. We here only give a brief summary of it. Instead of determining similar images to a given query using only one fixed weight set, we determine the *top-ranked* image (the nearest neighbour = NN) for *all possible* weight sets (given  $k$  features each associated with a weight, this requires a scan of a  $k$  dimensional vector space which can be done very efficiently using a recursive scan of an integer lattice imposed on the weight space). We call this set of top-ranked images the  $NN^k$  of

a query. The idea is that by not restricting ourselves to one particular weight set, we are able to capture and expose more of the different meanings an image may have. For each nearest neighbour, we record the proportion of the weight space for which it was ranked top. This provides us with a new measure of similarity which will be used in this and the subsequent visualization. We also record the average of all the weights for which a particular  $NN^k$  was ranked top.

Determining the  $NN^k$  of a query image forms the first step of the technique. The user then gives positive relevance feedback on the set of  $NN^k$ . The second step involves retrieving with the weight sets associated with the selected  $NN^k$  (the average weight sets mentioned above). The idea is that those weight sets will be likely to cause other similar images to surface. If more than one image has been selected, the ranked lists obtained for the different weights sets are merged. We compared performance of the two-step  $NN^k$  search with our own relevance feedback technique [5] and an alternative weight update method by Rui [10] using a small subset of the Corel Gallery 380,000 collection. The results (Figure 2) suggest that the  $NN^k$  technique does indeed hold some promise as a way of inferring feature weights at run-time.



**Fig. 2.** Mean average precision (MAP) plotted against number of categories for five different retrieval strategies each after one iteration of relevance feedback. Category size is set to 5 and thus the difficulty of the retrieval task increases with the number of categories. Our two-step technique consistently outperforms our own regression method (SSE)[5] as well as Rui’s method (RUI) [10]. BL is a baseline obtained by using equal weights for all features. OPT is the optimum performance obtained by using the best weight set for each query

## 5.2 Visualization and Interaction

For this visualization, we have taken particular care to avoid overlap between images. The results of the two steps are visualized in a similar fashion although the ways the displayed images are obtained differ significantly as we described above. The first step determines the set of  $NN^k$ , each with an associated weight set and a similarity value that is proportional to the number of weight sets for

which it was ranked top. The layout of images is achieved as follows: images are fitted on the screen in order of decreasing similarity to the query. The first image is displayed in the center, each subsequent image is placed at the first available position that does not produce an overlap. This position is determined by moving outwards from the center such that we trace out an Archimedean spiral. The first position thus found may not be optimal in the sense that the image may be moved still closer towards the center. To achieve a more compact layout, each image is therefore shifted from this initial position towards the center whilst no overlap occurs. Since the number of  $NN^k$  can vary considerably with the number of features used and the kind of query image, the size of the images cannot be fixed *a priori*. To ensure that all images fit on the screen, we start with an initially large image diameter and repeat the above procedure with a progressively smaller image diameter until we achieve a complete fit. This adjustment is made whenever the user resizes the window typically resulting in a different configuration of images. This display focusses the user's attention on those images that are more likely to be relevant, with those images that are ranked top for only a small proportion of weight combinations displayed not only in the periphery but also at a correspondingly smaller size. To view peripheral images more clearly, the user can drag any image closer towards the center, where it will be displayed at the same scale as the image the mouse arrow currently points at. The user may now inspect the set of  $NN^k$  and, using a pull-down menu, select the most relevant images either to expand the query, or to retrieve with the weight set associated with the selected images.

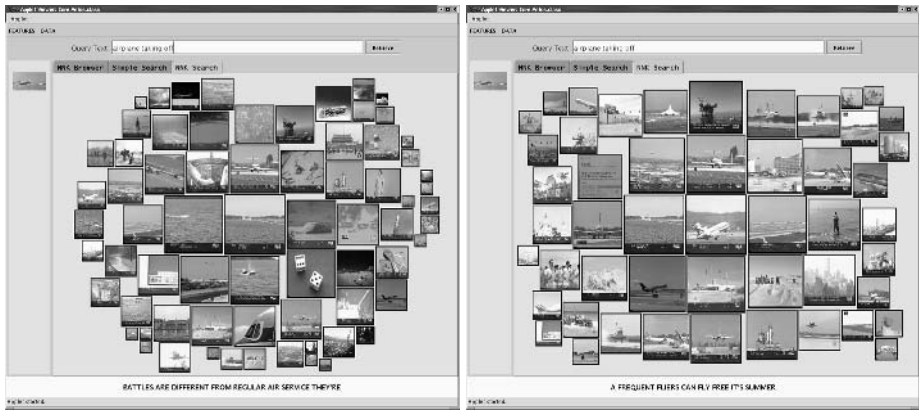
We illustrate the technique using topic 04 from TRECVID 2003 which asks for images depicting planes taking off. Query images are shown on the query canvas. The left picture in Figure 3 shows the  $NN^k$  for the query images with 7 planes among them, 2 of which appearing to take off. The user selects the image to the very left, shown in the center of the right screenshot. The number of planes has doubled with 5 out of 14 appearing to take off.

## 6 $NN^k$ Networks

### 6.1 Using $NN^k$ for Browsing

It is a natural extension of the  $NN^k$  search technique to allow the user to repeat the first step of the  $NN^k$  search and determine the  $NN^k$  for any of the displayed images (instead of asking for the ranked list produced when using the newly found weight set). This leads to the idea of an  $NN^k$  network where the vertices represent individual images and arcs are established between two vertices if one is the  $NN^k$  of the other, that is if there is at least one weight set for which one image is the nearest neighbour of the other. Again, we record for each nearest neighbour, the proportion of the weight space for which it was top-ranked.

The advantages of the proposed structure are threefold: first, by looking at a multitude of feature combinations, the network helps expose the semantic richness of images. It is left to any particular user to decide which of the possible interpretations is the most appropriate one by following the corresponding



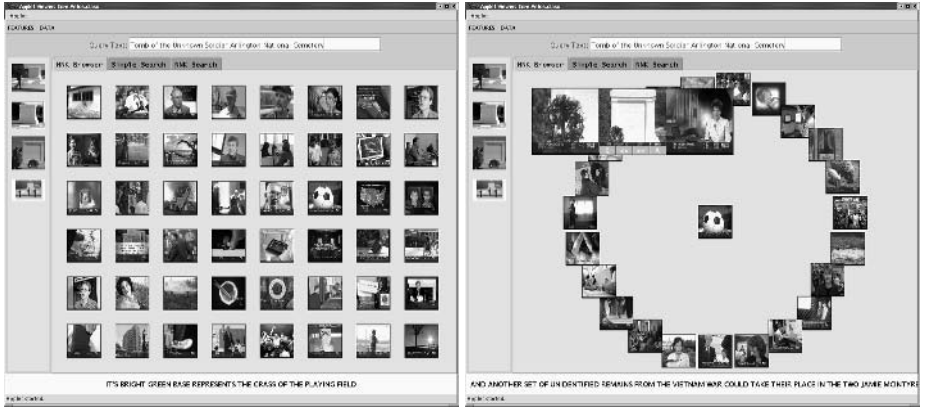
**Fig. 3.** The display of the first and second step of the  $NN^k$  search. The left figure displays the  $NN^k$  for the query images shown on the left canvas (planes taking off). The size of each image is proportional to the number of weights for which it was ranked top. The image north-west of the center has been selected as a relevant. The right canvas shows the improved results obtained when querying with the weight set associated with the selected image. See text for details.

outgoing arc in the network. Secondly, image access can be achieved without formulating a query. A mental representation of the image is sufficient to guide the user through the network. Thirdly, the network structure is entirely precomputed which allows interaction to take place in real time, regardless of the size of the collection represented by the network. In addition, we show in [6] that the resulting networks have desirable topological properties, including small-world properties (high clustering coefficient and small average distance between nodes) as defined in [14] and scale-freeness of the vertex degrees, suggesting that it is a structure which lends itself particularly well for browsing.

## 6.2 Network Visualization and Interaction

To provide initial access points for a user who does not want to formulate a query, we determine the nodes with the highest vertex outdegree. These nodes constitute hubs of the network that allow the user to reach deep into the structure along one or two arcs. The initial display can be seen on the left picture of Figure 4. If the user has formulated a query, we display not the set of high-connectivity nodes as initial access points but the results of the query, that is the same set of images displayed along the spiral in the simple search visualization. Browsing the network is achieved by clicking on any of the displayed nodes. This recovers the set of nearest neighbours from the database, which are then displayed on the screen such that their distances to the center are proportional to their dissimilarity to the selected node (where, again, we use the proportion of weight space for which the  $NN^k$  comes top as the measure of similarity).

We shall illustrate the usefulness of the browsing structure using topic 06 asking for images of the “Unknown Soldiers’ monument” in Arlington. As the query images show, the monument itself is of white colour, has a very distinctive shape and is set against a relatively dark background. The image from among the high-connectivity nodes that seems visually most similar is the football in the upper left corner on the left picture of Figure 4. Clicking on this image results in the display shown on the right with the football image itself placed in the center and its  $NN^k$  displayed around it. The enlarged image in the top right depicts the monument we are looking for (to the left and the right of that  $NN^k$  are shown the neighbouring key frames in the corresponding video sequence).



**Fig. 4.** The left screenshot shows the initial display of high-connectivity hubs of the  $NN^k$  network. The right screenshot has the selected image placed in the center and its  $NN^k$  arranged around it. Already at this stage we find that one of these images is relevant to the query.

The network structure was used extensively for TRECVID 2003 and proved instrumental for the success of the interactive runs. In one interactive run we restricted interaction to browsing only, and although the performance remained below that of other runs that employed some form of query-by-example, it was comparable to a large number of other interactive runs submitted by the other participants, and was significantly above the performance of our automated search run that used a fixed set of weights without any further user interaction (for details see [4]).

## 7 Conclusions

We have presented three different techniques for content-based access to image collections and described different visualization for each. There is quantitative evidence suggesting that the two-step  $NN^k$  search improves on traditional relevance feedback techniques for weight update, while the  $NN^k$  network appears to

fare very well as a complement to the traditional query-by-example paradigm. To exploit the full potential of these techniques, however, it is pivotal to tie them to efficient, user-friendly interfaces with a rich set of user interactions. This paper has proposed possible ways how this can be achieved in an integrated retrieval system.

**Acknowledgements.** This work was partially supported by the EPSRC, UK.

## References

1. G Aggarwal, T V Ashwin, and S Ghosal. An image retrieval system with automatic query modification. *IEEE Transactions on multimedia*, 4(2):201–213, 2002.
2. I Campbell. *The ostensive model of developing information-needs*. PhD thesis, University of Glasgow, 2000.
3. M Flickner, H Sawhney, W Niblack, Q H J Ashley, B Dom, M Gorkani, J Hafner, D Lee, D Petkovic, D Steele, and P Yanker. Query by image and video content: the QBIC system. *IEEE Computer*, 9:23–32, 1995.
4. D C Heesch, M Pickering, A Yavlinsky, and S Rüger. Video retrieval within a browsing framework using keyframes. In *Proceedings of TRECVID 2003, NIST (Gaithersburg, MD, Nov 2003)*, 2004.
5. D C Heesch and S Rüger. Performance boosting with three mouse clicks — Relevance feedback for CBIR. In *Proceedings of the European Conference on IR Research 2003*. LNCS, Springer, 2003.
6. D C Heesch and S Rüger.  $NN^k$  networks for content based image retrieval. In *Proceedings of the European Conference on IR Research 2004*. LNCS, Springer, 2004.
7. D C Heesch, A Yavlinsky, and S Rüger. Performance comparison between different similarity models for CBIR with relevance feedback. In *Proceedings of the International Conference on video and image retrieval (CIVR 2003), Urbana-Champaign, Illinois*. LNCS, Springer, 2003.
8. Klaus Mainzer. *Computerphilosophie*. Junius Verlag, 2003.
9. H Müller, W Müller, D M Squire, M S Marchand-Maillet, and T Pun. Strategies for positive and negative relevance feedback in image retrieval. In *Proceedings of the 15th International Conference on Pattern Recognition (ICPR 2000), IEEE, Barcelona, Spain*, 2000.
10. T S Rui, T S Huang, M Ortega, and S Mehrota. Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 123–131, 1998.
11. S Santini, A Gupta, and R Jain. Emergent semantics through interaction in image databases. *IEEE transactions on knowledge and data engineering*, 13(3):337–351, 2001.
12. Q Tian, B Moghaddam, and T S Huang. Display optimization for image browsing. In *International Workshop on Multimedia Databases and Image Communications*, 2001.
13. R S Torres, C G Silva, C B Medeiros, and H V Rocha. Visual structures for image browsing. In *Conference on Information Knowledge Management (CIKM'03)*, 2003.
14. D J Watts and S H Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

# Retrieving ClipArt Images by Content\*

Manuel J. Fonseca, B. Barroso, P. Ribeiro, and Joaquim A. Jorge

Department of Information Systems and Computer Science  
INESC-ID/IST/Technical University of Lisbon  
R. Alves Redol, 9, 1000-029 Lisboa, Portugal  
Fax: +351.21.3145843  
`{mjf,jaj}@inesc-id.pt, {bamb,pdsr}@mega.ist.utl.pt`

**Abstract.** Nowadays there are a lot of vector drawings available for inclusion into documents, which tend to be achieved and accessed by categories. However, to find a drawing among hundreds of thousands is not easy. While text-driven attempts at classifying image data have been recently supplemented with query-by-image content, these have been developed for bitmap-type data and cannot handle vectorial information. In this paper we present an approach to index and retrieve ClipArt images by content, using topological and geometric information automatically extracted from drawings. Additionally, we introduce a set of simplification heuristics to eliminate redundant information and useless elements. Preliminary usability tests to our prototype show promising results and suggest good acceptance of sketching as a query mechanism by users.

## 1 Introduction

Currently there are a huge number of drawings that users can integrate into their documents. However, to use one of those images, they have to browse through large and deep file directories or navigate a complex maze of categories previously defined to organize drawings. Furthermore, such search becomes humanly impossible when the number of drawings increases. One possible solution is to manually catalog all drawings by adding textual descriptions. However, this approach is not satisfactory, because it forces users to know in detail the meta-data used to characterize drawings. Yong Rui [1] analyzed several content-based image retrieval systems that use color and texture as main features to describe image content. On the other hand, vector drawings are represented in structured form requiring different approaches from image-based methods.

In the past years there have been some research works in retrieving drawings. Gross' Electronic Cocktail Napkin [2] addressed a visual retrieval scheme based on diagrams, to indexing databases of architectural drawings. Berchtold's S3 system [3] supports managing and retrieving industrial CAD parts, through

---

\* This work was funded in part by the Portuguese Foundation for Science and Technology, project 34672/99 and the European Commission, project SmartSketches IST-2000-28169.



contour matching. Park’s approach [4] retrieves mechanical parts based on dominant shapes and spatial relationships. Leung proposed a sketch retrieval method [5] for general unstructured free-form hand-drawings.

We can observe two things from existing content-based retrieval systems for drawings. The first is scalability: most published works use databases with few elements (less than 100). The second is complexity: drawings stored in the database are simple elements not representing sets of real drawings, such as ClipArt images.

We will now describe our approach to retrieve drawings by content privileging the use of spatial relationships and geometric information. Moreover, we perform automatic simplification, classification and indexation of existing drawings, to make the retrieval process both more effective and accurate. Additionally, fast and efficient algorithms to perform similarity matching between sketched queries and a large database of ClipArt drawings are required. Finally, we implemented a prototype to retrieve WMF ClipArts and performed usability tests, which show very encouraging results and suggest good acceptance of sketching by users.

## 2 Our Approach to Retrieve Vector Drawings

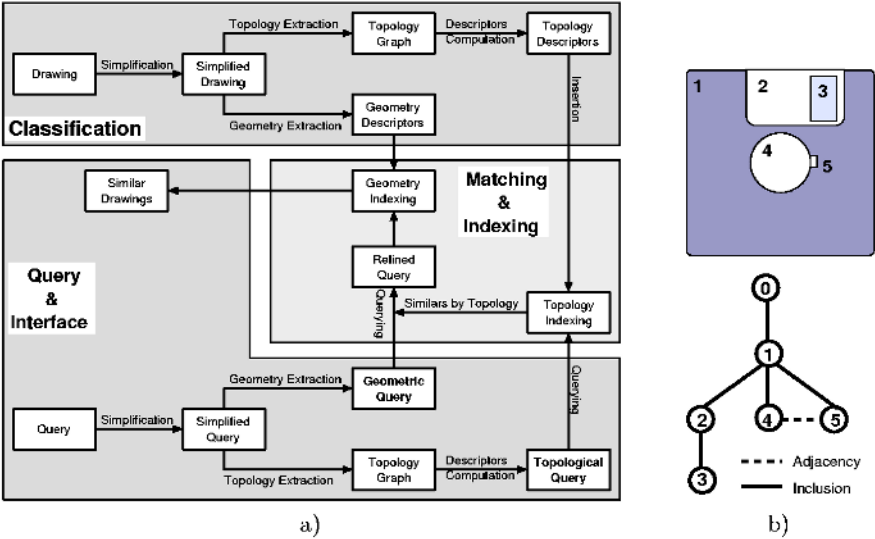
Our approach solves both scalability and complexity problems by developing mechanisms for retrieving drawings, in electronic format through hand-sketched queries, taking advantage of user’s natural ability at sketching and drawing. Moreover, unlike the majority of existing systems, our method was developed to support large sets of drawings. To that end, we devised a multidimensional indexing structure that scales well with growing data set size. Figure 1.a shows our system architecture, identifying its main components.

### 2.1 Classification

Content-based retrieval of pictorial data, such as digital images, drawings or graphics, uses features extracted from the corresponding picture. Typically, two kinds of features are used. Visual features encode information, such as color, texture and shape. Relationship features describe topological and spatial relationships among objects in a picture. However, for vectorial drawings, color and texture are irrelevant features. We focus on topology and geometry.

Our classification process starts by applying a simplification step, to eliminate most useless elements. The majority of ClipArt drawings contains many details, which are not necessary for a visual query and increase the cost of searching. We try to remove visual details (i.e. small-scale features) while retaining the perceptually dominant elements and shapes in a drawing. The main goal of this step is to reduce the number of entities to analyze in subsequent steps of the classification process, in order to speed up queries.

After simplification we identify visual elements, namely polygons and lines, and extract shape and topological information from drawings. We use two relationships, **Inclusion** and **Adjacency**, which are a simplified subset of the



**Fig. 1.** a) System architecture for our approach. b) ClipArt drawing (top) and correspondent topology graph (bottom).

topological relationships defined by Egenhofer [6]. Relationships thus extracted are compiled in a Topology Graph, where "parent" edges mean **Inclusion** and "sibling" connections mean **Adjacency**, as illustrated in Figure 1.b. While these relationships are weakly discriminating, they do not change with rotation and translation.

However, topology graphs are not directly used for searching similar drawings, since graph matching is a NP-complete problem. We use the corresponding graph spectra instead. For each topology graph to be indexed in a database we compute descriptors based on its spectrum [7]. In this way, we reduce the problem of isomorphism between topology graphs to computing distances between descriptors. To support partial drawing matches, we also compute descriptors for sub-graphs of the main graph. Moreover, we use a new way to describe drawings hierarchically, by dividing them in different levels of detail [8] and then computing descriptors at each level. This combination of sub-graphs descriptors and levels of detail, provides a powerful way to describe and search both for drawings or sub-parts of drawings, which is a novel feature of our work.

To acquire geometric information about drawings we use a general shape recognition library called CALI [9]. This enables us to use either drawing data or sketches as input, which is a desirable feature of our system. We use CALI to compute a set of geometric features such as area and perimeter ratios from special polygons such as the convex hull, the largest area triangle inscribed in the convex hull or the smallest area enclosing rectangle, among others. Using geometric features instead of polygon classification, allows us to index and store poten-

tially unlimited families of shapes. Experimental evaluation [10] revealed that this technique outperforms other methods to describe shapes, such as Fourier descriptors, grid-based descriptors or Delaunay triangulation, yielding better precision figures for all recall values. We obtain a complete description of geometry in a drawing, by applying this method to each geometric entity of the figure. The geometry and topology descriptors thus computed are inserted in two different indexing structures, one for topological information and another for geometric information, respectively.

## 2.2 Query and Matching

Our system includes a Calligraphic Interface to support the specification of hand-sketched queries, to supplement and overcoming limitations of conventional textual methods. The query component performs the same steps as the classification process, namely simplification, topological and geometric feature extraction, topology graph creation and descriptor computation. This symmetrical approach is unique to our method. In an elegant fashion two types of information (vector drawings + sketches) are processed by the same pipeline.

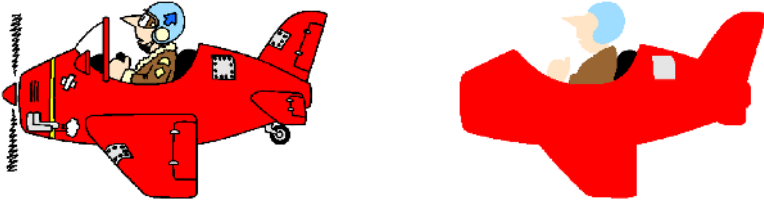
We developed a new multidimensional indexing structure, the NB-Tree [11], which provides an efficient indexing mechanism for high-dimensional data points. The NB-Tree is a simple, yet efficient indexing structure, using dimension reduction. It maps multidimensional points to a 1D line by computing their Euclidean Norm. In a second step we sort these points using a  $B^+$ -Tree on which we perform all subsequent operations.

Computing the similarity between a hand-sketched query and all drawings in a database can entail prohibitive costs especially when we consider large sets of drawings. To speed up searching, we divide our matching scheme in a two-step procedure. First, we select a set of drawings topologically similar to the query, by performing a KNN query to the topology indexing structure. This step works as a filter, reducing the number of potential candidates to compare in the next step. Second, we use geometric information to further refine the set of candidates.

## 3 Simplification Heuristics

To simplify drawings we used a set of heuristics that explore their specific features and human perception. This reduces both the information present in drawings, storage space and processing time. We focus our heuristics in three particularities of ClipArt drawings: color gradients, contours and small area polygons.

*Color Gradient.* Many ClipArt drawings use continuous and overlapped polygons with small changes in color, to achieve a gradient effect. Since our approach describes drawings using only topology and geometry, color is not relevant for retrieval. However, we use color information to simplify drawings, by grouping polygons with similar colors into a single polygon.



**Fig. 2.** Application of Heuristic 3. Original (left) with 591 polygons and simplified (right) with 13 polygons.

*Contour Lines.* We found out that many shapes were defined using two polygons, one to specify the filled region and another just to define the contour. Since the second polygon do not convey any additional information, this heuristic goal is to eliminate them.

*Small Area Polygons.* ClipArt drawings have a lot of small area polygons to describe details that users ignore while specifying queries. This heuristic discards small polygons, when comparing to the largest one. The biggest challenge here was the definition of “small”. To that end we used several percentages of the biggest polygon during simplification and asked users to analyze the results. We also considered trade-offs between simplification and precision values.

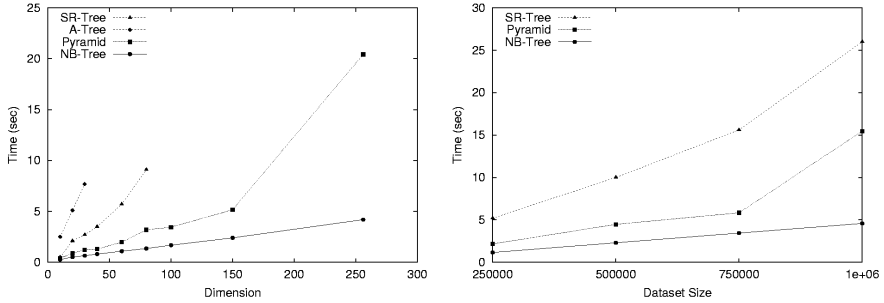
## 4 Evaluation and Experimental Results

Whereas the critical step in classification (using our approach) is drawing simplification, in nearest neighbor matching search dominates the resource usage.

*Simplification and Classification.* To determine the degree of simplification we applied the three heuristics to a set of 30 drawings randomly selected and we counted the number of polygons and lines before and after simplification. We found out that for this set we achieved a simplification degree of around 80%, on average, for lines and polygons. It is important to notice that after simplification, users still recognize drawings.

We also measured classification times on a AMD Duron @ 1.3GHz with 448MB of RAM, running Windows XP. We classified 968 drawings in 7 minutes and 55 seconds, yielding an average of 0.49 seconds per each drawing. This is the overall classification time, which includes simplification, geometric and topological feature extraction, descriptors computation and insertion in the indexing structures. The resulting indexing structures required a storage space of 16.8 MB (excluding drawings). We can consider that the classification process is fast and that the storage space required is relatively small, making this approach suitable for large data sets of drawings.

*Indexing Structure.* We shortly describe experimental comparison of our indexing structure (NB-Tree) to the most popular approaches available, such as the SR-Tree, the A-Tree and the Pyramid Technique. From Figure 3 we can see that the NB-Tree outperforms all the structures evaluated when data dimension and data set size increases. A more detailed evaluation can be found in [11].



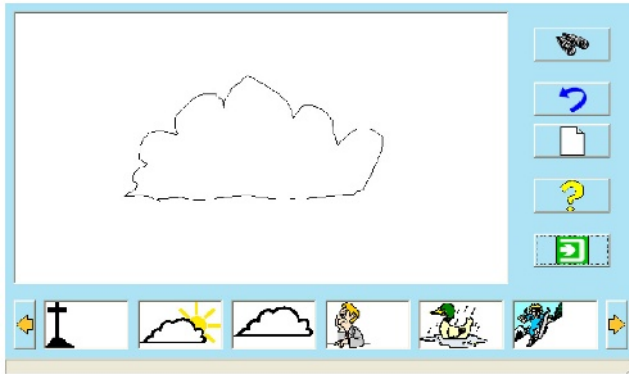
**Fig. 3.** Search times for K-NN as a function of dimension (left) and data set size (right).

*ClipArt Retrieval.* We developed a prototype to retrieve ClipArt drawings through hand-sketched queries (see Figure 4). On the top-left we can see the sketch of a cloud and on the bottom results returned by the implied query, ordered from left to right. We also provide a way to perform Query-by-Example allowing the user to select a result and use it as query.

In order to assess acceptance and recognition-level performance, we conducted preliminary usability tests involving twelve users and a database of 968 drawings from several categories. These drawings were classified using our hierarchical scheme to produce descriptors for each level of detail and for each subpart. Resulting descriptors were then inserted into two databases, one for topology and another for geometry, using our NB-Tree.

Task 1 comprises the searching of a drawing by providing a verbal description of objects. The main goal of this task was to measure user satisfaction about returned results when using sketches and Query-by-Example. Results revealed that searching using sketches was in general less successful than using Query-by-Examples. This is due mainly to people drawing skills. However, when users resorted to Query-by-Example functionality, they found results more satisfactory. Query times, using an AMD Duron @ 1.3GHz were between 1 and 2 seconds, which most users found satisfactory.

In Task 2 we asked users to search for drawings depicted on a paper and we checked in what position the corresponding drawing appeared in results. We observed that the best results were achieved for drawings containing collections of easy-to-draw shapes and with a strong topological component. In these cases, the topological filtering is effective and reduces the number of drawings to compare in the geometric matching. Furthermore, easy-to-draw shapes assure that users will sketch something very similar to the desired drawing.



**Fig. 4.** ClipArt finder prototype.

Finally, we collected users opinions through a questionnaire. Users liked the interaction paradigm very much (sketches as queries), were satisfied with returned results and pleased with the short time they had to spend to get what they wanted.

## 5 Conclusions and Future Work

We have presented a generic approach suitable for content-based retrieval of drawings. Our method hinges on recasting the general picture matching problem as an instance of graph matching using vector descriptors. To this end we index drawings using a *topology graph* which describes adjacency and containment relations for parts and subparts. We then transform these graphs into descriptor vectors in a way similar to hashing to obviate the need to perform costly graph-isomorphism computations over large databases, using spectral information from graphs. Finally, a novel approach to multidimensional indexing provides the means to efficiently retrieve sub-drawings that match a given query in terms of its topology. We described the overall process to simplify drawings using a set of heuristics and usability tests performed using our prototype to retrieve ClipArts. Users were generally pleased with both the returned drawings and using sketches as the main query mechanism. We are currently working towards converting this application into a Sketch-Based Web search engine for ClipArt drawings.

## References

1. Rui, Y., Huang, T.S., Chang, S.F.: Image Retrieval: Current Techniques, Promising Directions, and Open Issues. *Journal of VCIR* **10** (1999) 39–62
2. Gross, M., Do, E.: Demonstrating the Electronic Cocktail Napkin: a paper-like interface for early design. In: *Proceedings of CHI'96*. (1996) 5–6
3. Berchtold, S., Kriegel, H.P.: S3: Similarity in CAD Database Systems. In: *Proc. of the Int. Conference on Management of Data (SIGMOD'97)*. (1997)

4. Park, J., Um, B.: A New Approach to Similarity Retrieval of 2D Graphic Objects Based on Dominant Shapes. *Pattern Recognition Letters* **20** (1999) 591–616
5. Leung, W.H., Chen, T.: Hierarchical Matching for Retrieval of Hand-Drawn Sketches. In: *Proceedings of IEEE ICME'03*. (2003) 29–32
6. Egenhofer, M.J., Al-Taha, K.K.: Reasoning about Gradual Changes of Topological Relationships. Volume 639 of *LNCS*. Springer-Verlag (1992) 196–219
7. Cvetkovic, D., Rowlinson, P., Simic, S.: *Eigenspaces of Graphs*. Cambridge University Press, United Kingdom (1997)
8. Fonseca, M.J., Jr., A.F., Jorge, J.A.: Content-Based Retrieval of Technical Drawings. *Special Issue of IJCAT* (to appear) (2004)
9. Fonseca, M.J., Jorge, J.A.: Experimental Evaluation of an on-line Scribble Recognizer. *Pattern Recognition Letters* **22** (2001) 1311–1319
10. Fonseca, M.J., Barroso, B., Ribeiro, P., Jorge, J.A.: Retrieving Vector Graphics Using Sketches. In: *Proc. of the Smartgraphics Symposium'04* (to appear). (2004)
11. Fonseca, M.J., Jorge, J.A.: Indexing High-Dimensional Data for Content-Based Retrieval in Large Databases. In: *Proceedings of DASFAA'03*. (2003) 267–274

# Use of Image Subset Features in Image Retrieval with Self-Organizing Maps<sup>\*</sup>

Markus Koskela, Jorma Laaksonen, and Erkki Oja

Laboratory of Computer and Information Science, Helsinki University of Technology  
P.O.BOX 5400, FI-02015 HUT, FINLAND  
{markus.koskela,jorma.laaksonen,erkki.oja}@hut.fi

**Abstract.** In content-based image retrieval (CBIR), the images in a database are indexed on the basis of low-level statistical features that can be automatically derived from the images. Due to the semantic gap, the performance of CBIR systems often remains quite modest especially on broad image domains. One method for improving the results is to incorporate automatic image classification methods to the CBIR system. The resulting subsets can be indexed separately with features suitable for those particular images or used to limit an image query only to certain promising image subsets. In this paper, a method for supporting different types of image subsets within a generic framework based on multiple parallel Self-Organizing Maps and binary clusterings is presented.

## 1 Introduction

Content-based image retrieval (CBIR) addresses the problem of finding images relevant to the users' information needs from image databases, based principally on low-level visual features for which automatic extraction methods are available. Due to the semantic gap, i.e. the inherently weak connection between the high-level semantic concepts that humans naturally associate with images and the low-level features that the computer is relying upon, the task of developing this kind of systems is very challenging. One popular method to improve retrieval performance is to use relevance feedback [1], i.e. to adjust the subsequent retrieval process by using information gathered from the user's intra-query feedback.

Another approach to improve retrieval results is to group somehow similar images together and use these groupings to filter out a portion of the non-relevant images for a given query. Unfortunately, in many applications, no semantic annotations or categorizations exist and they are difficult to produce automatically. Still, producing low-level classifications and, in some cases, also certain semantic categorizations are possible with current automatic methods. Examples of low-level classification are distinguishing photographs from computer-generated

---

<sup>\*</sup> This work was supported by the Academy of Finland in the projects *Neural methods in information retrieval based on automatic content analysis and relevance feedback* and *New information processing principles*, the latter being part of the Finnish Centre of Excellence Programme.



graphics [2,3] and separating color and grayscale images. Certain types of semantic image categories can be distinguished with specialized classifiers which typically perform two-class classifications to the database images. This type of image classification has been studied, for example, to distinguish indoor and outdoor images [4], city images from landscape scenes [5], man-made vs. natural environments [6] and for portraits vs. non-portraits [3]. The effort needed for manual annotation can also be reduced e.g. with active learning [7].

Additionally, such a categorization can be useful in limiting feature extraction to images which are suitable for that particular method. For example, extracting color features may be appropriate only to color images, and shape features requiring segmentation are valid for images containing salient objects and not e.g. for landscape or textural images. A further example is face detection and recognition: in addition to being an important cue of the semantic content in itself, a detected face also makes it viable to extract specific features for face recognition from that image.

In this paper, we extend our existing CBIR system structure to support subset features and binary classifications in addition to the database-wide features used in earlier work. As the main indexing method, we use the Self-Organizing Map (SOM) [8] and propose a uniform framework for incorporating all these feature types into a single system and utilizing them simultaneously and in parallel in image queries. The paper is organized as follows. Section 2 discusses different types of indices needed when some image classification methods and image subsets produced with them are available. In Section 3, the SOM is presented as a common tool for a variety of indices. A set of experiments in which previously recorded user interaction is used as an example case is presented in Section 4. Section 5 then concludes the paper.

## 2 Indexing Image Subsets

In this section, we consider feature extraction and indexing methods for a whole database and for different types of image subsets. To begin with, it is convenient to identify two fundamental subset types. First, the relevant information of a subset can be contained in set membership, i.e. the subset consists of images having a specific property; or second, the subset contains images for which a certain feature extraction method is either meaningful or available. In the latter case, the mere set membership is semantically insignificant and the pertinent information lies in the internal structure of the subset.

These two types of subsets are clearly not contradictory, but often highly correlated. For example, an object detection module can be used to find images representing a salient object in the foreground and some shape-based features can then be used to describe these images; without prior object detection the feature is useless. On the other hand, whether a photograph is in color or grayscale has often little effect on the semantic content, but e.g. hue-based features give meaningful results only for color images. According to this viewpoint, we can recognize the following three types of image indices.

## 2.1 Full Indices

Since object and scene recognition and semantic classification in heterogeneous image databases are very difficult problems, the basic features in CBIR are typically extracted from the whole database. Since only a little can be assumed about the image content and the features are to be automatically extracted, such features are usually limited to low-level statistical representations such as global color and texture features.

## 2.2 Binary Classifications

With some image categorization methods at disposal, we can obtain partitionings of the image database. Commonly, these methods yield two-class classifications for the images; an image either contains or lacks a certain characteristic. A set of different classifications can then be combined to construct a binary index.

An important example of binary classification indices for heterogeneous images is keyword annotation. A straightforward way to utilize keywords describing image contents is to use them as binary attributes affixed to the images. Each keyword divides the database into two subsets: images having and not having that specific keyword in their annotations.

## 2.3 Image Subsets with Internal Structure

An image subset may also consist of images for which a given feature extraction method is relevant. Extracting these features from all images may be a waste of computational resources or even harmful to retrieval performance. Therefore, such subsets need to be indexed separately. The same holds for situations where a certain feature is available only for a fraction of the database.

One method to gradually improve the retrieval performance of CBIR systems is to utilize information provided by relevance feedback in an inter-query learning scheme. The fact that two images received similar relevance assessments during a specific query is a cue for similarities in their semantic content. With a limited number of recorded query sessions, only a subset of the database has probably been processed in these queries. Still, even a relatively small number of stored queries can improve retrieval results considerably [9]. A similar setting takes place when a portion of the database has been annotated with some accuracy, whereas the remaining images do not contain these annotations.

# 3 Self-Organizing Map as an Indexing Framework

In indexing methods based on clustering, the data is divided into clusters with the intention that only one or a few of these clusters have to be exhaustively processed in one given query. Typically, each cluster is represented by its centroid or a representative data item and, instead of the original data, the query is first compared to the centroids or cluster representatives. The best cluster or clusters

according to the used similarity measure are then selected and the data items belonging to those clusters are evaluated in full. Finally, a fixed number of the most similar items are returned as the result of the query.

This basic clustering approach can readily be extended for our purposes as follows. First, we allow clusterings to be only partial, i.e. there may be data items that do not belong to any cluster, and, second, a data item is allowed to belong to more than one cluster. A set of binary classifications can then be represented with the same data structure, although the individual clusters are now formed of images sharing a certain attribute instead of applying some unsupervised clustering algorithm acting on automatically extracted low-level features.

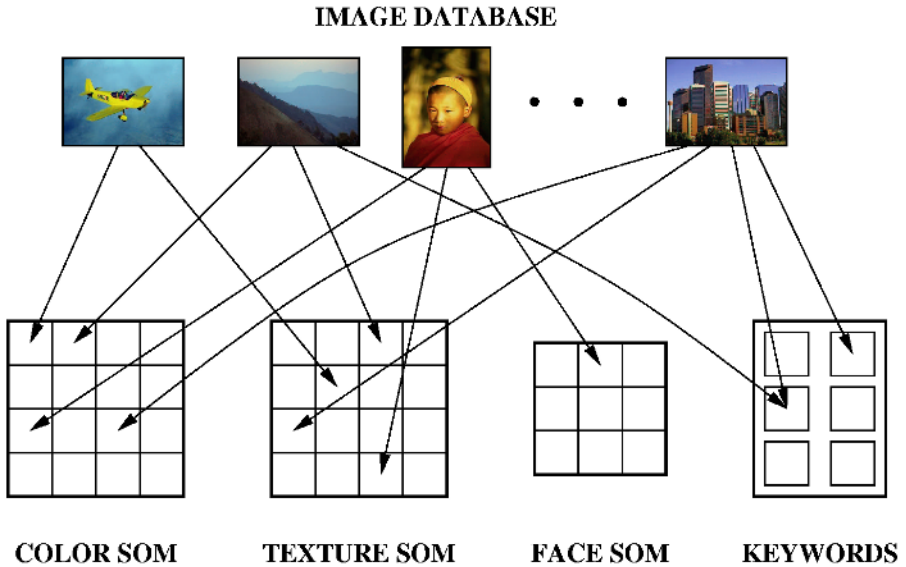
The Self-Organizing Map (SOM) [8] can also be considered as a clustering method due to the mapping of feature vectors and their associated images to the SOM units. This, however, ignores the topology of the SOM, so a portion of the provided data organization is dismissed. In fact, the distinct strength of the SOM as an indexing method lies in its property of topology preservation. The SOM preserves topology on the map grid and this enables the spreading of the user-provided relevance assessments also to the neighboring map units since they can be assumed to contain similar feature vectors and thus similar images.

### 3.1 PicSOM

The PicSOM [10,11] image retrieval system is a framework for research on content-based image retrieval. As the name implies, PicSOM uses the SOM as its basic image indexing method, although other clustering methods are also supported. For example,  $k$ -means clustering was experimented with and compared to the SOM in [11]. Instead of the standard SOM version, PicSOM uses a special form of the algorithm, the Tree Structured Self-Organizing Map (TS-SOM) [12]. The hierarchical structure of TS-SOM is useful for two purposes. First, it drastically reduces the complexity of training large SOMs needed for indexing large databases by exploiting the hierarchy in finding the best-matching map unit (BMU) for an input vector. Second, the hierarchical representation of the image database produced by a TS-SOM can be utilized in browsing and visualizing the images in the database.

### 3.2 Multiple Self-Organizing Maps

The PicSOM system is fundamentally based on using several parallel SOMs trained with separate feature data simultaneously in image retrieval. The features are usually comprised of statistical visual data such as the MPEG-7 [13] content descriptors. Any additional vectorial data can, however, be used to train corresponding SOMs and thus be used in image retrieval. If the feature in question was extracted only from a subset of the images in the database, only that subset is used in the training the corresponding SOM. The size of the SOM should be set accordingly, as it is not reasonable to use SOMs of the same size with small subsets as with the whole database.

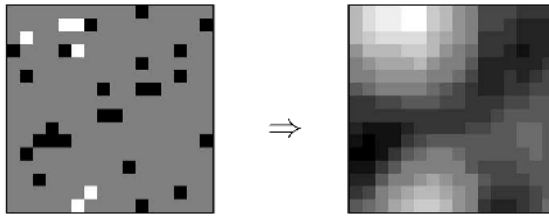


**Fig. 1.** An example of using four parallel indices for an image database. The color and texture SOMs are trained with the whole database and the face SOM with a subset obtained with face detection. Keyword annotations are also available for some images.

After training the SOMs, their map units are connected with the images of the database. This is done by locating the BMU for each image on each SOM. As a result, the different SOMs impose different similarity relations on the images and the system thus inherently uses multiple features for image retrieval. An illustration with two full-database SOMs, one SOM trained with an image subset, and keyword-based binary classifications is presented in Figure 1.

### 3.3 Relevance Feedback with Multiple Feature Indices

The relevance feedback mechanism of PicSOM is a crucial element of the retrieval engine. The basic method is only briefly presented here, see [10] for a comprehensive treatment. During a retrieval session, the user marks images that she considers relevant as positive, and the remaining ones are implicitly regarded as negative. As the first step, the SOM units are awarded a positive score for every relevant image mapped in them resulting in an attached positive impulse. Likewise, associated non-relevant images result in negative scores and impulses. If the total numbers of relevant and non-relevant shown images are  $N^+(n, m)$  and  $N^-(n, m)$  at query round  $n$  on  $m$ th SOM, the positive and negative scores are simply the inverses:  $x_+(n, m) = 1/N^+(n, m)$  and  $x_-(n, m) = -1/N^-(n, m)$ . For each SOM, these values are mapped from the shown images (and thus rated either as positive or negative) to their corresponding BMUs where they are then summed. This way, we obtain a zero-sum sparse value field on every SOM in use. With SOMs trained with image subsets, we neglect the shown images that are



**Fig. 2.** An example of how a SOM surface is convolved with a tapered window function. On the left, images selected and rejected by the user are shown with white and black marks, respectively. On the right, the convolution result, where relevance information is spread around the centers.

not mapped to that particular SOM. Since the sparse value fields are zero-sum, we introduce no bias against the non-indexed images.

Due to the topology preservation of the SOM, we are motivated to spread the relevance information (both positive and negative) provided by the user also to the neighboring map units of the BMUs. This can be done by convolving the sparse value fields with tapered (e.g. triangular or gaussian) window functions. Figure 2 illustrates how the positive and negative responses are first mapped on a  $16 \times 16$ -sized SOM and how these responses are expanded in the convolution.

The indices formed by binary classifications are treated similarly with two exceptions. First, since no topology exists, the spreading of user responses with convolution is not valid (or, we always use unit impulse as the convolution window). Second, the same image may be present in multiple clusters, which is taken into account by dividing the relevance weight equally to all these binary classes.

As the response values of the parallel indices are mutually comparable, we can determine a global ordering for determining the overall best candidate images. By locating the corresponding images in all indices, we get their scores with respect to different feature extraction methods. The total qualification values for the candidate images are then obtained simply by summing the corresponding responses. Content descriptors that fail to coincide with the user's conceptions mix positive and negative user responses in the same or nearby map units and binary classes. Therefore, they produce lower qualification values than those descriptors that match the user's expectations and impression of image similarity and thus produce areas or clusters of high positive response. As a consequence, the parallel content descriptors and indices do not need explicit weighting.

## 4 Case Study: Recorded User Interaction

In the following experiments, we use a database of  $N = 59\,995$  miscellaneous images from Corel Photo CDs. We created manually the following six image classes as ground-truth: **faces** (1115 images, *a priori* probability 1.85%), **cars** (864 images, 1.44%), **planes** (292 images, 0.49%), **sunsets**, (663 images, 1.11%), **horses**, (486 images, 0.81%), and **traffic signs**, (123 images, 0.21%). As visual features, we used a subset of MPEG-7 [13] descriptors, viz. *Scalable Color*,

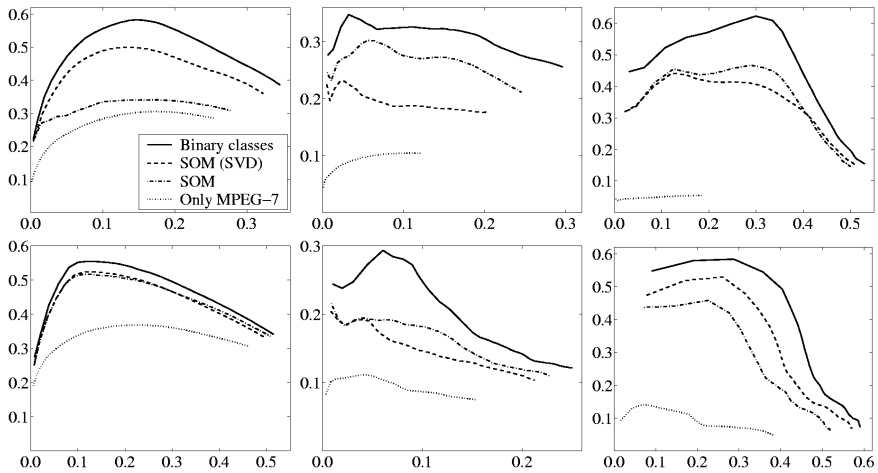
*Dominant Color, Color Structure, Color Layout, Edge Histogram, Homogeneous Texture, and Region Shape.* These descriptors were extracted from every image in the database, and  $256 \times 256$ -sized SOMs were then trained for each of them.

As an example case, we study the utilization of previously recorded relevance evaluations provided by users during earlier query sessions (inter-query learning). The relevance evaluations provided by the user during a query session partition the set of shown images into relevant and nonrelevant classes with respect to the target of that particular query. Specifically, when two images both have been marked as relevant within the same query, we can assume that the semantic contents of the images are somehow similar. In the experiments, the relevant images of individual queries are used as image clusters. The relevance evaluations consisted of 317 saved query sessions in which a total of 6897 images (11.5% of the database) had been marked relevant at least once. On the average, a recorded query contained 25.8 relevant images. As the first one of the studied methods, we use these image clusters directly as a binary classification index.

Alternatively, we use the evaluations as statistical features by constructing a fixed-length binary feature vector for images with recorded evaluations so that each dimension of the vector corresponds to a recorded query. Since the remaining images do not have any stored assessments, they are omitted from this inter-query index. This way, we obtain 317-dimensional inter-query feature vectors for the 6897 images. Thirdly, as a preprocessing step, we reduce the dimensionality of these feature vectors to 50 with singular value decomposition (SVD). For comparison, we train two  $64 \times 64$ -sized SOMs for the feature vectors, one before and the other after the dimensionality reduction.

In the test setting, each image in the studied six ground-truth classes was used one at a time as an initial reference image for category search. The system returned 20 images at each round, and with 50 rounds per query session the total number of shown images was  $N_T = 1000$  images, i.e. 1.67% of the database size. Relevance feedback was used to refine the query as categorical feedback, i.e. images belonging to the studied class were marked as relevant and others as nonrelevant. This way, the retrieval experiments could be carried out automatically. In spreading the responses of the sparse value fields, triangular windows of 6 and 8 map units in length were used for the user interaction SOMs and the other SOMs, respectively.

The averaged recall–precision plots for the six ground truth classes are shown in Figure 3. The MPEG-7 descriptors are used in all cases, either solely or with one of the three inter-query indices. Overall, it can be seen that due to the semantic gap, the precision of using the low-level features alone remains modest, especially since no segmentation was used. Using information provided by the recorded queries considerably improves retrieval precision regardless of the used method, even though only 11.5% of the images are included at all in the recordings. Using the recorded queries as binary classifications yielded better results than their use as statistical feature vectors with SOMs. This, however, comes with an increase in computational requirements, since the shown images are often mapped to multiple clusters. Also, the approach does not scale well to



**Fig. 3.** Recall-precision plots (x-axis: recall; y-axis: precision) using the MPEG-7 descriptors solely and with recorded queries as binary classifications and as subset features with and without dimensionality reduction (SVD). Used classes were (top row, left-to-right) **faces**, **cars**, **planes**, (bottom row) **sunsets**, **horses**, and **traffic signs**.

large amounts of recorded data since the number of binary classification clusters equals the number of recorded queries.

The use of a subset feature SOM based on the recorded queries also improves the results significantly. With this approach, the increase in online computational requirements is minor since we only add one SOM index along the seven parallel SOMs trained with the MPEG-7 descriptors. Reducing the dimensionality of the inter-query feature by SVD before the SOM training does not systematically alter the retrieval results; with **faces** and **traffic signs** the dimensionality reduction improves results, with **cars**, **planes**, and **horses** the results are worse. In an application where a lot of recorded usage data could easily be accumulated, this kind of preliminary dimensionality reduction would be essential as the dimensionality of the training data equals the number of recorded image queries, which could well be in the order of thousands or more.

## 5 Conclusions and Future Work

With large databases of general images, the retrieval performance of low-level visual features alone often remains quite modest (as is observed also in Fig. 3) and additional feature types can be highly beneficial. One method for improving results is to incorporate some automatic image classification methods to the retrieval system. The resulting subsets can then be indexed separately or used to limit the query only to specific image subsets.

In the experiments of this paper, previously recorded query sessions were interpreted either as binary classifications or as statistical features for an image

subset and used in parallel with visual MPEG-7 descriptors. The experiments showed that both approaches produced clearly improved results and that using this information greatly enhances the precision of the system without any additional burden to the user. While using previously stored retrieval sessions performed by assorted users of the system might conflict with the subjectivity and context-dependency of human notion of image similarity, in practice the user assessments provide valuable accumulated information about image semantics.

A straightforward direction for future work is to include automatic semantic classifications to the PicSOM system and study whether indexing these subsets separately would be beneficial. For this purpose, a generic framework of feature indices, both database-wide and subset-based, was presented in this paper.

## References

1. Zhou, X.S., Huang, T.S.: Relevance feedback for image retrieval: A comprehensive review. *Multimedia Systems* **8** (2003) 536–544
2. Frankel, C., Swain, M.J., Athitsos, V.: Webseer: An image search engine for the world wide web. Technical Report 96-14, The University of Chicago (1996)
3. Gevers, T., Aldershoff, F., Geusebroek, J.M.: Integrating visual and textual cues for image classification. In: *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France (2000) 419–429
4. Szummer, M., Picard, R.W.: Indoor-outdoor image classification. In: *Proceedings of IEEE International Workshop on Content-Based Access of Image and Video Database*, Bombay, India (1998) 42–51
5. Vailaya, A., Jain, A., Zhang, H.J.: On image classification: City images vs. landscapes. *Pattern Recognition* **31** (1998) 1921–1935
6. Yoon, J., Jayant, N.: Semantics-sensitive image retrieval: An information fusion approach. In: *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2003)*. Volume 1., Baltimore, MD, USA (2003) 761–764
7. Sychay, G., Chang, E., Goh, K.: Effective image annotation via active learning. In: *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002)*. Volume 1., Lausanne, Switzerland (2002) 209–212
8. Kohonen, T.: *Self-Organizing Maps*. Third edn. Springer-Verlag (2001)
9. Koskela, M., Laaksonen, J.: Using long-term learning to improve efficiency of content-based image retrieval. In: *Proc. of Third International Workshop on Pattern Recognition in Information Systems (PRIS 2003)*, Angers, France (2003) 72–79
10. Laaksonen, J., Koskela, M., Laakso, S., Oja, E.: Self-organizing maps as a relevance feedback technique in content-based image retrieval. *Pattern Analysis & Applications* **4** (2001) 140–152
11. Laaksonen, J., Koskela, M., Oja, E.: PicSOM—Self-organizing image retrieval with MPEG-7 content descriptions. *IEEE Transactions on Neural Networks, Special Issue on Intelligent Multimedia Processing* **13** (2002) 841–853
12. Koikkalainen, P., Oja, E.: Self-organizing hierarchical feature maps. In: *Proceedings of International Joint Conference on Neural Networks*. Volume II., San Diego, CA, USA (1990) 279–284
13. MPEG: MPEG-7 Overview vers. 9 (2003) ISO/IEC JTC1/SC29/WG11 N5525.



# An Indexing Model of Remote Sensing Images

Paola Carrara<sup>1</sup>, Gabriella Pasi<sup>2</sup>, Monica Pepe<sup>1</sup>, and Anna Rampini<sup>1</sup>

<sup>1</sup> IREA-CNR, Via Bassini 15, I-20133 Milan (Italy)

<sup>2</sup> ITC-CNR, Via Bassini 15, I-20133 Milan (Italy)

**Abstract.** This paper is concerned with the problem of indexing remote sensing images. This kind of images has a semantics mainly related to the image spectral properties. For these reasons the spectral properties can be considered as effective image descriptors. The model proposed in this paper assumes that the image descriptors are spectral regions and their spectral signatures. By the application of a clustering algorithm each image is segmented into a set of spectral regions to be associated to basic (pre-defined) ground cover classes. To take into account the uncertainty that often affects the cluster labelling process the indexing model generates for each region and each reference class a possibility degree indicating the possibility that the region corresponds to that class. The uncertainty of this association is explicitly modelled, and allows the definition of a more flexible image representation with respect to a crisp approach.

## 1 Introduction

An efficient management of huge collections of remote sensing images and the development of effective retrieval mechanisms are becoming an important need. The traditional approach to manage and retrieve remote sensing images has been for long time based on the association of meta-information with each image, with the consequent limitations of reduced scalability and difficulty to identify the meta-information itself. To overcome these limitations some Content-Based Image Retrieval (CBIR) methods have been proposed for remote sensing (RS) images. Remote sensing imagery is a narrow domain of application, where the variability of images is limited and the semantic description is well-defined. It has been observed [9] that for such domain the gap between features and their semantic interpretation (the so-called *semantic gap*) is usually small, so domain-specific models may be of help. Some CBIR approaches to remote sensing images are therefore based on meaningful domain-dependent features, i.e. the spectral properties of the images [1][6][10][2].

In this paper an indexing model of remote sensing images is proposed, which employs as meaningful features the image spectral properties. It is assumed that the atomic components of the considered information items (i.e., remote sensing images) are pixels, represented as vectors of values (called the *spectral signatures*). For a given archive of remote sensing images, a set of meaningful ground cover classes (for example: vegetation, water, bare soil, etc.) is defined ‘a priori’.

Each image is clustered into a set of significant patterns (homogeneous regions) which are associated to the ground cover classes. The association of an image region with a predefined, meaningful ground cover class, is performed by evaluating the similarity of the region signature and the spectral signatures of the reference ground cover classes.

The process of associating the image regions with the reference ground classes is characterized by uncertainty, which is usually not taken into account in the indexing process. For example, in [1] the spectral properties allow to segment images in regions to which more indexes are associated, representing both spectral and non-spectral aspects of the regions; [6] performs a clustering of the images and associates the clusters to classes by exploiting expert knowledge, while in [10] the association process is based on statistic considerations. Also the approach in [2] is based on the extraction of clusters, but they are not explicitly associated to ground cover classes. The main novelty of the approach proposed in this paper consists in explicitly modelling the uncertainty intrinsic both in the extraction of significant patterns (homogeneous regions) from the images and in their assignment to ground cover classes. This uncertainty is represented by means of the framework of possibility theory [4][11]: for each region of an image the possibility degree of being a ground class is computed on the basis of qualitative decision rules expressed on the similarity measures. An image is therefore formally represented as a set of regions, each of which is associated with a possibility distribution over the reference classes.

It has been noticed [5] that in the CBIR field we need techniques that allow to reason even from incomplete information: this paper addresses the problem of measuring the similarity also when some information lack or many different integrated aspects have to be taken into account. The intent of this research is not to contribute to remote sensing image understanding, but to adapt well performing techniques in this area to develop better storage and retrieval approaches. The proposed indexing method has been tested on an archive of Landsat satellite imagery of Italy by the creation of a spectral library derived from other Landsat scenes.

The paper is organized as follows: in Sect. 2 the spectral properties of images are analyzed and the conditions under which they can be proposed as image descriptors are discussed. In Sect. 3 the indexing model is defined and the data structure as well as some ideas concerning the definition of the query language are sketched.

## 2 Spectral Properties as Descriptors of RS Images

In remote sensing images the energy reflected from the earth surface is digitally acquired. For each spectral band, each pixel of the satellite image is therefore represented by a Digital Number (DN) in a range that depends on the radiometric resolution of the sensor, i.e. the number of bits used. In this work we deal with two sensors - Thematic Mapper (TM) and Enhanced Thematic Mapper plus (ETM<sup>+</sup>)- on board of Landsat 5 and 7 satellites, which acquire in 8 bits

the energy reflected from the Earth surface in six spectral bands from visible to shortwave infrared (TM1, TM2, TM3, TM4, TM5, TM7), and emitted in one thermal infrared band (TM6). Each surface type is therefore characterised by a vector of DN<sub>s</sub>, called the spectral signature, with seven elements in the range [0..255]. Each surface type is characterised by peaks in absorption and reflection: though some variations can occur, the overall shape of the signature is generally maintained. Some macro-classes, such as vegetation, bare soil, water, clouds, snow, may be easily distinguished on the basis of their reflective properties. For example, the vegetation has a remarkably high reflection in the near infrared region of the electromagnetic spectrum, and a low reflection in the visible region corresponding to red. This allows to distinguish vegetation areas from bare ground as the difference of reflection in TM3 and TM4 is greater for vegetated areas than for bare ground (see the first and second graphs in Fig.1 respectively). On the contrary, different types of vegetation, or different conditions of growth of the same type of vegetation, present similar spectral responses with small variations, as it can be observed in the first graph of Fig.1 showing many curves belonging to the macro-class vegetation.

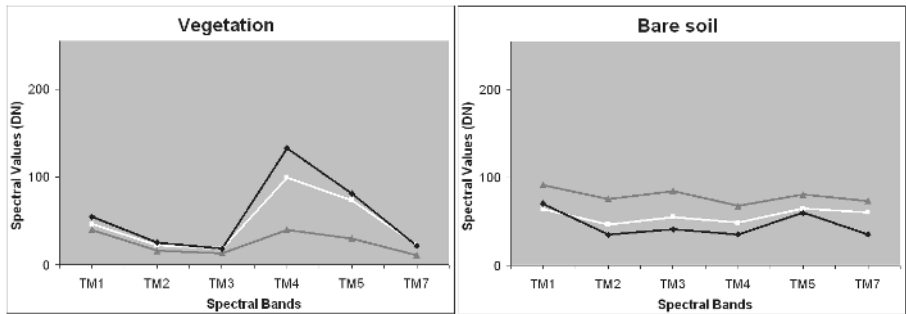
The extraction of descriptors from remote sensing images is performed by means of a classification process, which is based on the use of a set of pre-defined ground cover classes. Multi-spectral classification is an information extraction process that analyzes the spectral signatures and then groups pixels into regions having similar signatures; it can be performed by unsupervised or supervised techniques depending on the available 'a priori' information [8].

The supervised classification is based on the use of 'a priori' knowledge called ground truth, gathered on a geographically representative sample of objects and used to train the classifier. When the 'a priori' knowledge lacks, an unsupervised classification can be used to group multi-spectral response patterns in separable clusters to be then labelled into ground-cover classes by an 'a posteriori' interpretation.

In the context of automatic indexing the classification process has to be as far as possible accurate without requiring any specific human knowledge, since it has to be used to represent and retrieve information for heterogeneous users with unknown abilities and background, and, it has to be independent on the increase and extension of the data volume [2]. The proposed solution consists therefore in the adoption of an unsupervised clustering approach followed by labelling of identified clusters based on the knowledge of reference spectral responses [3]. The accuracy of the association of clusters with reference classes depends both on a meaningful collection of representative spectral signatures and on the definition of suitable similarity measures discussed in the next session.

### 3 An Indexing Model of Remote Sensing Images

As outlined in the previous section, the definition of the image representation is accomplished through a complete image classification process, performed in subsequent steps, which are synthesized in Table 1. The first phase of the clas-



**Fig. 1.** Graphic representations of spectral responses in digital number values of vegetation (left), and bare soil (right); on the x-axis we reported six reflective bands on the Landsat satellites

**Table 1.** Steps of image classification

- Step 1 - Extraction of cluster centers:** The K-Means algorithm extracts a given number of cluster centers (the number has been heuristically set to 20 with a maximum number of iterations set to 15); this technique is applied on elements randomly selected all over the image. It then assigns all the image pixels to the identified clusters.
- Step 2 - Measures of similarity:** SMC and CC measures between the spectral signatures identifying the cluster centroids and the reference spectral signatures are computed.
- Step 3 - Labelling:** Two heuristic tests based on the SMC and CC measures are evaluated to associate clusters to reference classes (with a possibility degree).

sification process aims at identifying in each image a set of spectral regions. The subsequent steps aim at ‘recognizing’ the identified regions, i.e. at associating with each of them one or more of the predefined ground cover classes. The usual approaches to classify remote sensing images by spectral signatures perform this association in a crisp way, by selecting just one class for each region, based on the application of a similarity measure. This process induces a loss of information: in fact, the crisp selection of a ground label may cause interpretation errors, due either to the mixed nature of certain regions or the variability of the image acquisition conditions. In order to achieve a better classification and a more faithful representation of images, more information is required in the formal image representation. To this aim, the proposed indexing model defines an image representation which incorporates the information of uncertainty in the association of an image region with the ground cover classes. As in certain cases a spectral region may present similarity with more than one cover class, the similarity values are used as a basis for expressing the uncertainty in this association. From a formal point of view this is modelled through possi-

bility theory, which constitute a suitable framework for managing uncertainty in knowledge representation. The similarity of an image region with a ground cover class is interpreted as the possibility that the region corresponds to the considered ground class. Each image region is therefore represented by means of a spectral signature with associated a possibility distribution over the ground classes. In the following subsections the formal aspects of the indexing procedure are described.

### 3.1 Defining the Image Representation

To obtain the formal representation of a remote sensing image, the first step consists in the image clustering: each pixel in an image belongs to a spectral region, which groups a set of homogeneous pixels, and which is also identified by a representative spectral signature. In this paper the K-Means clustering algorithm [8] has been adopted. Its output is a set of clusters characterized by the spectral values of their centroids. The subsequent steps compare the spectral signature of each region with the spectral signatures of the reference classes, to the aim of identifying to which class each cluster corresponds. To make this comparison, a similarity measure has to be adopted. Among the several similarity measures proposed in the literature [7], we have selected two measures, i.e. the Simple Matching Coefficient (SMC)(see equation 1), which quantifies the similarity of vectors' values, and the Correlation Coefficient (CC)(see equation 2), which quantifies the similarity of trends represented by vectors. This choice has been done to take into account two distinct aspects of similarity between spectral signatures, i.e. a value-to-value similarity and a shape similarity.

$$s_{smc}(\bar{x}_i, \bar{x}_j) = 1 - \frac{\sum_k |x_{ik} - x_{jk}|}{mN_k} \quad (1)$$

$$N_k = \max_{i,j} |x_{ik} - x_{jk}| \quad \text{with } 1 \leq k \leq m$$

where  $\bar{x}_i, \bar{x}_j$  are vectors of values  $\in [0..255]$ , and  $m$  is the number of spectral bands.

$$s_{cc}(\bar{x}_i, \bar{x}_j) = \frac{\sigma_{ij}}{\sigma_i \cdot \sigma_j} \quad \text{with } -1 \leq s_{cc} \leq 1 \quad (2)$$

where  $\sigma$  is the covariance.

As previously outlined, the SMC and CC measures refer to distinct aspects of the similarity between two vectors, and they are defined on distinct numerical domains. Moreover, for distinct ground cover classes, the same SMC and CC values (computed for a given image region) may correspond to different levels of similarity. For these reasons, to the aim of both expressing similarity based on SMC and on CC on the same scale, and adapting to each ground class the interpretation of similarity based on numerical SMC and CC values, for each

ground class two compatibility functions are heuristically defined. A first function refers to the SMC similarity measure; it takes values on the domain of SMC values and produces a value in the interval  $[0,1]$ . The second function refers to the CC similarity measure; it takes values on the domain of CC values and produces a value in the interval  $[0,1]$ . The value produced by these functions can be interpreted as the compatibility between a numerical measure and an ‘acceptable’ similarity with the considered reference class. Formally, the compatibility functions can be interpreted as the membership functions of fuzzy subsets on the SMC and CC numerical domains respectively. We denote these functions by  $\mu_{SMC}$  and  $\mu_{CC}$  respectively. In Fig.2 an example of membership functions is showed for the classes vegetation and water respectively. From a formal point of view this filtering applied on the SMC and CC values for a global similarity assessment, makes the two distinct measures of similarity comparable (they are in fact defined on the same range), and their aggregation may be performed thus producing a global numeric degree which can be interpreted as a possibility degree.

Let us examine in more detail the decisional process which allows the definition of a possibility distribution for each image region. For each representative region in a given image, first of all the numeric values ( $smc_i$ ) and ( $cc_i$ ) are computed to assess the similarity to the  $i$ -th reference class (we suppose here that the reference classes are  $n$  and  $i \leq n$ ). Then the membership degrees  $\mu_{SMC}(smc_i)$  and  $\mu_{CC}(cc_i)$  are computed and aggregated to obtain an overall degree of similarity, which is interpreted as the possibility that the considered image region corresponds to the reference class  $i$ . This aggregation is performed by applying the minimum operator:

$$p_i = \text{Min}(\mu_{SMC}(smc_i), \mu_{CC}(cc_i))$$

The minimum performs a pessimistic aggregation of the two considered value; other aggregation (for example mean) operators could be applied, they will be analyzed in future developments of this work. This process is applied for each reference class, so that for each spectral region a possibility distribution is computed and stored.

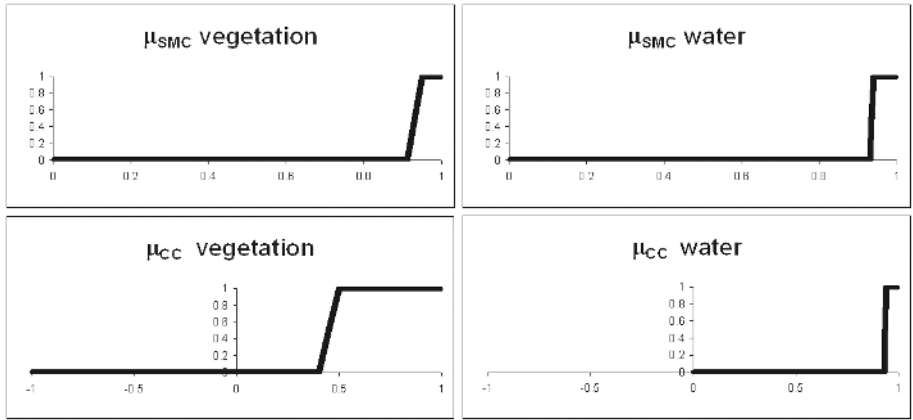
### 3.2 Data Structure and Query Language

The indexing process guides to the definition of a suitable data structure, able to store the information extracted from the indexed images. We imagined a traditional IR design with a data structure constituted of two main components: the dictionary of the spectral signatures of the reference ground classes and the inverted file. Each element of the dictionary is constituted by the couple ( $\langle$ class name $\rangle$ ,  $\langle$ vector of values of the reference spectral signature $\rangle$ ) and points to the list of all the images containing the reference class  $\langle$ class name $\rangle$  (see Fig.3).

The inverted file contains the lists of the representations of the archived images; each list corresponds to a class of the dictionary. Each element of the list pointed by the  $i$ -th class is constituted by an archived image identifier  $I_j$ , a spectral region

identifier  $SR_{jk}$  as the image  $I_j$  may contain more than a  $SR$  referred to the same class, the vector of the centroid of the spectral region in the image itself  $VC_{jk}$ , the two computed similarities  $smc_{ijk}$  and  $cc_{ijk}$  and the degree of possibility  $p_{ijk}$  that the region  $k$  in image  $j$  is associated with the pointing class  $i$ .

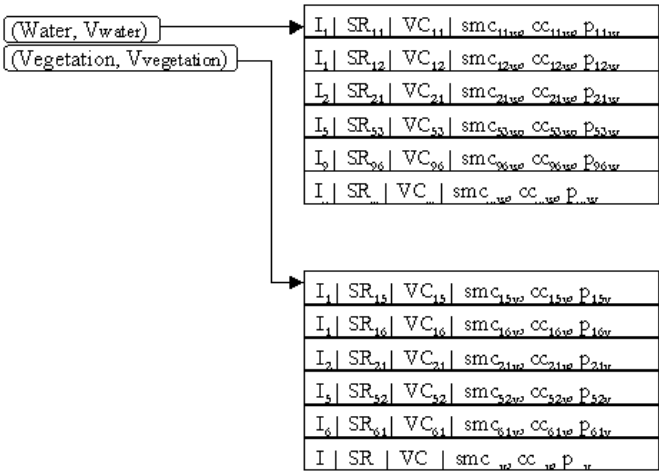
As the basic elements of the data structure are constituted by the reference classes - identified by their spectral signatures - the query language allows to express constraints on these information elements. We can propose three main types of class specifications: by textual identifier (query by label), by spectral signature (query by vector of spectral values) and by image sub-window (query by example). In all the above cases the result of a query evaluation should be a list of images with an associated relevance estimate (the so called RSV). The RSV can be computed on the basis of the possibility degrees  $p_i$  of the image clusters pointed by the reference class  $i$  identified by the query.



**Fig. 2.** SMC membership functions for the class Vegetation (upper left) and for the class Water (upper right); CC membership functions for the class Vegetation (lower left) and for the class Water (lower right)

## 4 Conclusions

The indexing model proposed has been evaluated by an experimental data set of Landsat TM images containing 263 subimages of 81 km<sup>2</sup> each. They record heterogeneous scenes of the Italian peninsula in different seasons and the total covered area is 18,747 km<sup>2</sup>. Landscapes range from coastal zones to mountain regions, as well as agricultural and urban areas; cloud covered areas are also included. This heterogeneity improves the quality of the experiment. The collection of spectral signatures has also been created including general purpose ground cover classes: vegetation, water, bare soil/urban, snow, cloud, shadow. The reference signatures has been based on mean spectral signatures collected from images which are not included in the archive: the main selection criteria



**Fig. 3.** An illustration of the data structure; the boxes on the left are two elements of the dictionary, while on the right there are the pointed lists

are statistical parameters and the representation power. The experiments proved that a good collection of reference signatures is a key factor in the obtained results.

The evaluation has been performed applying the indexing method to all the image archive and verifying its performance with respect to four predefined classes: vegetation, water, cloud and bare soil. The encouraging results of the indexing are shown in Table 4. Note that the high number of false positives for the cloud class is due to misjudgement with the class snow, which has not been considered in the experiment.

**Table 2.** Indexing results: first row reports the number of images actually containing the class; the second row images where the class has been identified by the indexing procedure; last rows reports the percentage of false negatives and false positives respectively

Class $i$	Vegetation	Water	Cloud	Bare soil
Truth <sub><math>i</math></sub>	230	218	67	231
Indexing <sub><math>i</math></sub>	230	226	84	234
False negatives <sub><math>i</math></sub>	0	2	3	0
False positives <sub><math>i</math></sub>	0	10	20	3

Among the future developments we plan to model also spatial characteristics of images. The spectral regions identified by the indexing procedure correspond to several objects spatially distributed within an image. The spatial features in terms of metric and topological relationships of the identified objects could then



be included. This extension enables the definition of a richer query language, allowing the specification of spatial constraints.

## References

1. Barros J, French J, Martin W, Kelly P, White JM (1994) Indexing Multispectral Images for Content-Based Retrieval. In: *Image and Information Systems: Applications and Opportunities (23rd AIPR Workshop)*, Proc. SPIE 2368, Washington DC, Oct. 1994, 25–36
2. Bretschneider T, Cavet R, Cao O (2002) Retrieval of remotely sensed imagery using spectral information content. In: *Proceedings of the International Geoscience and Remote Sensing Symposium*, 4:2253–2256
3. Carrara P, Galli C, Rampini A (2000) A Database for Remote Sensing Image Retrieval by Spectral Features. ITIM-CNR Tech. Rep., September 2000
4. Dubois D, Prade H (1988) *Possibility Theory: An Approach to Computerized Processing of Uncertainty*. Plenum Press, New York
5. Eakins JP (2002) Towards intelligent image retrieval, *Pattern Recognition*, 35:3–14
6. Koperski K, Marchisio GB (2000) Multi-level Indexing and GIS Enhanced Learning for Satellite Imagery. In the *Proceedings of the Workshop on Multimedia Data Mining MDM/KDD2000*, August 20-23, 2000 Boston (MA) USA, 8–13
7. Miyamoto S (1990) *Fuzzy Sets in Information Retrieval and Cluster Analysis*. Kluwer Academic Publishers, Dordrecht
8. Schowengerdt RA (1997) *Models and Methods for Image Processing*, Academic Press, San Diego
9. Smeulders AWM, Worring M, Santini S, Gupta A, Jain R (2000) Content-Based Image Retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(12):1349–1380
10. Vellaikal A, Kuo CC, Dao S (1995) Content-Based Retrieval of Remote Sensed Images Using Vector Quantization. In: *Proceedings of SPIE Visual Information Processing* 2488:178–189
11. Zadeh LA (1978) Fuzzy Sets as a Basis for a Theory of Possibility. *Fuzzy Sets and Systems*, 1:3–28

# Ambient Intelligence Through Image Retrieval

Jean-Marc Seigneur<sup>1</sup>, Daniel Solis<sup>2</sup>, and Fergal Shevlin<sup>1</sup>

<sup>1</sup>Department of Computer Science, Trinity College Dublin,  
Jean-Marc.Seigneur@trustcomp.org, Fergal.Shevlin@cs.tcd.ie

<sup>2</sup>Facultat d'Informàtica de Barcelona, Universitat Politècnica de Catalunya  
solisand@tcd.ie

**Abstract.** An ambient intelligent environment needs dynamic enrollment of strangers without too much human intervention. For this purpose, we propose an entity recognition process based on images captured with low-cost but widespread webcams and easy-to-deploy image processing techniques. We find that the use of levels of confidence in recognition due to different techniques and context-based image retrieval improves the process.

## 1 Introduction to Ambient Intelligence

Weiser's vision of ubiquitous computing [14] will be realised when computing capabilities are woven into the fabric of everyday life. Already companies in the domestic appliance market are promoting their smart home appliances – with communication, computation and data storage capabilities. Through such initiatives, spaces will become smart: endowed with ambient intelligence (AmI) to enhance the user's experience. However, challenges remain for the fulfilment of this vision. These include auto-configuration and autonomy, especially with respect to security [8]. The solution may come from a “concierge” process aware of what happens in the space, which can recognise strangers, acquaintances, friends or foes.

Vision is an obvious mechanism for the recognition of people in spaces. It has been used for authentication based on visual biometrics (such as fingerprint, face or gait recognition [1, 6, 10]). Generally, these techniques are used in controlled environments, where enrollment is mandatory (i.e., persons to be enrolled have their visual biometrics entered into the security system in advance). This contrasts with the fundamental requirement for ubiquitous computing environments: to allow for potential interaction with unknown entities [9]. In an AmI environment, enrollment cannot always require human intervention, e.g., from a system administrator. A smart space is not an improvement if it makes busy householders even busier. In public environments, there is no list of known people to be enrolled. People roam from one space to another as they wish. This introduces the requirement for smooth dynamic enrollment, i.e., the door should not be closed to strangers, but instead any stranger presenting themselves might become an acquaintance. To allow for dynamic enrollment of strangers and unknown entities, we have proposed an entity recognition (ER) process [9].

In this paper we investigate image retrieval as part of an ER scheme, called the vision entity recognition scheme (VER). We use commercial-off-the-shelf (COTS) products (e.g., basic “webcam” shipped with PC) in order to get an idea of what could be ubiquitously achievable today. The ER process allows the use of any scheme (i.e.,

even weak or unreliable) by taking into account confidence in recognition. We investigate how to improve indexing and retrieval of previously recorded imagery based on its context (e.g., time and weekday) in addition to its content.

## 2 Applying Vision Techniques to ER: VER

One of the foundations of security is authentication. Stajano [12] emphasized that without being sure with whom an entity interacts, the three fundamental properties – confidentiality, integrity and availability – can be trivially violated. Usually, authentication is the first step to ensure security in computing environments but other work [3] discusses why traditional authentication should be reconsidered for pervasive computing. Our end-to-end trust model [9] addresses this problem by starting with recognition, which is a more general concept than authentication, i.e., entity recognition encompasses authentication. To allow for dynamic enrollment of strangers and unknown entities, we have proposed the entity recognition (ER) process, which consists of four steps:

1. Triggering of the recognition mechanism
2. Detective Work to recognize the entity using the available recognition scheme(s)
3. Discriminative Retention of information relevant for possible recall or recognition
4. Upper-level Action based on the outcome of recognition, which includes a level of confidence in recognition

As an example of what is possible with this approach, we have developed an entity recognition component based on pluggable recognition modules (PRM), which allows the integration of more or less secure recognition schemes. We conjecture that the ability to recognise another entity, possibly using any of its observable attributes, is sufficient to establish trust in that entity based on past experience. The “Resurrecting Duckling” security policy model [13] is an example of entity recognition; ducklings know that their mother is the first entity who sent the imprinting key when they were born. They must be able to recognise when the entity with which they interact is the one who sent the imprinting key, no more. Most of the ER schemes cannot be considered as strong recognition schemes. However, we can still use them in our recognition process since the outcome of the ER process provides meta-data on the level of confidence in recognition including technical trust in the recognition scheme used.

In our current prototype, we assume a room with one door (see Fig. 2) and the equipment described in the Appendix. We combine different image processing and retrieval techniques (discussed in Section 3 and Section 4) to recognise people entering and leaving the room. The ER process allows recognition of previously observed/encountered entities based on visual evidence, i.e., imagery. There is a PRM where different vision schemes can be implemented (e.g., face matching or clothes colour). Each time someone moves in front of the camera, the ER process (depicted in Fig. 1) is triggered: we call this self-triggering because the system itself takes the initiative to start the recognition process in order to recognize potential surrounding entities. In step 2 of the ER process, the detective work consists of carrying out a variety of visual analyses to obtain a level of confidence of each recognition.

Retrieval of previous imagery is based on content as well as context (see Section 4). Step 3 is closely related to step 2 because discriminative retention of recognition must be based on previously stored imagery. A difficult question is to define when the person who enters the room is new and converge to the real number of different persons monitored so far. In the ER process, there is no initial mandatory enrollment but enrollment is moved down in the process and occurs at step 3 when recognition information on a new entity is stored for the first time and for later recall. A person is digitally represented by a principal ( $p$ ). The indexing of stored imagery for future retrieval at the end of step 3 also makes use of context (see Section 4). Step 4 of the ER process concerns further actions to be taken according to what person is recognised. This is almost beyond the scope of the paper but it may also be used to increase the level of work to be done during one round of ER. For example, if a new person is recognised at 2am, the concierge should increase its level of suspicion (and maybe send a warning message to the security guards) as well as increase the level of detective work and discriminative retention (which may augment the chance to later recognise the potential theft).

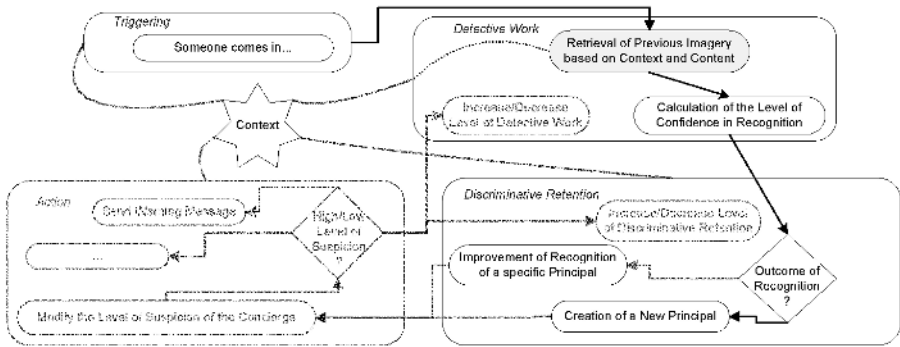


Fig. 1. VER process diagram

### 3 Image Processing Techniques

Due to the important requirement that the system needs as little as possible set-up or calibration by the owner of the space, the techniques used for image segmentation and analysis are necessarily simple. Additionally, the near-real time performance requirements of the system preclude the analysis of complex biometric characteristics such as gait, but we have designed our indexing and retrieval scheme to allow the inclusion of such characteristics should sufficient computational power exist.

#### 3.1 Feature Segmentation

Simple inter-frame image subtraction allows motion to be identified. If the motion blob area exceeds a certain threshold then it is considered a potential person. The region-merging via boundary-melting algorithm [11] is applied to segment the blob into distinct regions of significance for recognition. The significant regions, as shown in Fig. 2, are:

- a) Skin. Using the approach to skin segmentation suggested by Perez et al. [7], we transform from (R,G,B) colour space into the normalised (RN,GN) model and classify a pixel as skin if its values lie between certain upper and lower thresholds in RN and GN.
- b) Face. The uppermost region of skin exceeding a certain area threshold and with appropriate elongation is considered to be the face. Its bounding rectangular region extracted. If the region is larger or smaller than 40x40 pixels then it is sub- or super-sampled as appropriate to facilitate inter-image comparison.
- c) Clothes. Non-skin regions exceeding a certain area are considered clothes. There are typically two such regions found: top and bottom.
- d) Hair. In theory it should be relatively straightforward to segment hair, using its colour as another feature to facilitate recognition. However in our environment was insufficient contrast between the hair and the background for it to be segmented reliably.
- e) Height. Relative height can be approximated as the difference between the highest and lowest segmented pixel. Any height comparison must take into account the position of the feet.



**Fig. 2.** Environment and segmented features

### 3.2 Feature Analysis

The face is the only feature that can be used for recognition with any reasonably high degree of confidence (as shown in Subsection 5.2). To date we have used simple template-matching (normalised cross-correlation) to match segmented faces. However a principal components analysis (PCA) approach could also be used and would probably yield better results, as has been shown in [5].

## 4 Context-Based Image Retrieval

Each time a face is segmented from the real-time video sequence, it is appended to a list. When the sequence is finished, each face of the list is compared to the set of

different segmented faces stored previously. If there is no match above a minimal level of confidence, or no faces have been stored previously, it is added to the list of observed faces. Details of the other segmented features (for example, clothes colour) are associated with the face, as are temporal attributes such as date, time, and day of week. If a face matches above the minimal level of confidence then the other details are retrieved and used in the recognition process. In our approach, there is no training data and database of known users per se due to the requirement of dynamic enrollment. This differs from related work on real-time vision-based multi-modal recognition [10].

The advantage of pervasive computing environments is that computing entities are context-aware – environmental information that is part of an application's operating environment can be sensed by the application. Castro and Muntz [2] pioneered the use of context for multimedia object retrieval. We apply the concept within our ER process, which enables the concierge to adapt retrieval and recognition based on context and level of suspicion without the help of an administrator. Crowley et al.'s software architecture [4] for observing and modelling human activity built on top of their ontology for context-awareness is valuable for our type of application scenario. In their approach, our concierge may be seen as a supervisory controller of the ER process, which corresponds to an entity grouping of observational processes. Dey defines context as “any information that can be used to characterize situation” [4]. We especially make use of time and date to index and retrieve imagery.

#### 4.1 For Indexing

The first time the VER scheme is started in a new space, the list of faces and associated visual and temporal attributes is empty. As soon as someone comes in front of the camera, a sequence of faces is extracted from the video. Associated with each sequence is a structure storing the other elements of specific context. Our proof-of-concept implementation consists of storing the time and the day of the week. For each sequence, height and colour information is also computed. A database is used to store the recognition clues extracted from each sequence. These recognition clues are indexed in specific rows and each row consists of a supposed different person. We can then dynamically index the different rows based on context similarity. For example, we can order the rows decreasingly from the row which contains images the most often seen on Monday mornings around 8am. For performance reasons, each sequence of images is processed for face template matching after the end of the sequence when nobody is moving in front of the camera. The face template matching process is too expensive to be run in parallel during the capture of the sequence. Once all the images of the sequence are compared, we obtain a probability distribution of the following form:

$$(N_{PFRi} + N_{FRI}) * p_1 + \dots + (N_{PFRi} + N_{FRI}) * p_i + \dots + (N_{PFRn} + N_{FRn}) * p_n + N_{unknown} * p_{unknown} + N_{discarded} * p_{discarded}$$

where  $p_i$  is the supposed different person  $i$  among  $n$  previously seen persons,  $N_{PFRi}$  is the number of perfect face recognitions (match above PerfectFaceRecognition) of person  $i$ ,  $N_{FRI}$  is the number of face recognitions (match below PerfectFaceRecognition but above KnowFaceRecognition) for the person  $i$ ,  $N_{unknown}$  is the number of faces either of a new person or a very different face profile of a known person (match below KnowFaceRecognition but above BogusFaceDiscarded) and  $N_{discarded}$  is the

number of images considered to be of bad quality (below `BogusFaceDiscarded`). From this distribution, a choice has to be made. Is it a new person or should it update the recognition clues of a previously known person? The update only consists of faces that are considered different enough that previous images (that is, between `PerfectFaceRecognition` and `BogusFaceDiscarded`) in order to improve scalability. In cases where face recognition confidence is borderline, we use the other visual attributes to help in the decision-making process. So far, we have followed an empirical solution, which has given encouraging results in real settings. However, we have chosen to discard sequences which might pollute the database with poor quality face images. The simplified pseudo-code of the algorithm is depicted below:

```
Pick the person i with the greatest (NPFRI+NFRI)
if (NPFRI>(10%*TotalOfNotDiscardedImages))
    UpdateFacesOfPersoni
if (NoPerfectMatch)
{
    if (NFRI>50%*Nunknown)
        if (((HeightMatching*50%)+(ColourMatching*50%)) >= 50%)
            UpdateFacesOfPersoni
    if (Nunknown>50%*NFRI)
        if (((HeightMatching*50%)+(ColourMatching*50%)) < 50%)
            CreateNewPerson
}
```

## 4.2 For Retrieval

Thanks to our indexing, we can prioritize the retrieval based on context (i.e., time and day of the week). There are four parameters used in our algorithm: `TimeAndDayOfWeeK`; `PerfectFaceRecognition` (i.e., the percentage threshold above which the recognition match is considered perfect: empirically from the reading of several sequence processing samples say 92%), `UnknowFaceRecognition` (that is, the percentage threshold below which the recognition match is considered either a new person or a very different face profile of a known person: again empirically say 85%) and `BogusFaceDiscarded` (i.e., the percentage threshold below which the recognition match indicates that the image does not correspond to a face and is discarded: we empirically chose 30%). In order to benefit from a probabilistic approach and the fact that the images of a same sequence correspond to the same person (who is entering the room), at most 30 faces are extracted from the sequence and compared to all previous faces stored in the database. In order to speed up the process, the comparison is stopped if `PerfectFaceRecognition` is reached and the images stored in the database are reordered. The reordering consists of presenting the images of the previously recognised person first, ordered by their number of previous matches. Section 5.1 evaluates this retrieval approach.

## 5 Evaluation of the Retrieval System

We start by comparing random-based and context-enhanced retrieval and indexing. Then, we discuss which vision techniques are more important for improving the accuracy and relevance feedback of retrieval.

## 5.1 Context Versus Random for Retrieval and Indexing

One of the reasons we chose a context-based retrieval and indexing rather than retrieval with a random order of images is to obtain a faster retrieval scheme. It is worth mentioning that stopping the retrieval and not assessing all stored images for each new image is faster but we lose the opportunity to detect a recognition result greater than the PerfectFaceRecognition. However, this allows us to compare if context-based is really faster than random-based retrieval.

For this assessment, videos of 10 different persons (including Europeans, Chinese and Indians) entering the room 4 to 5 times were recorded. The database was populated with the same sequence for each person: these 10 sequences resulted in 205 faces stored in the database. Then, the remaining sequences of each person were processed (although no update/creation was applied) and resulted in the extraction of 757 faces. Using a random approach, this corresponds to 155185 matches ( $757 \times 205$ ). Thanks to our PerfectFaceRecognition bound and context reordering (explained in Subsection 4.2), assuming that each match takes the same time, the process was roughly 1.4 faster (that is, 44780 fewer matches were needed).

## 5.2 Empirical Assessment of the Technical Trust of Each Vision Technique

Each recognition scheme has to be assessed concerning its technical trustworthiness, which can be seen as the relevance feedback of retrieval. The number of people we used for this assessment (i.e., 10) is in line with the assessment done in previous related work [10] (i.e., 12).

The outcome of the ER process can be a set of  $n$  principals ( $p$ ) associated with a level of confidence in recognition ( $lcr$ ). The VER scheme is proactive: it triggers itself and uses a range of vision techniques which give evidence to compute a probability distribution of recognised entities. For example, a person among  $n$  persons previously recognised enters a room which is equipped with a biometric ER scheme. The outcome of recognition demonstrates hesitation between two persons:  $p_2$  and  $p_3$  are recognized at 45% and 55% respectively, these percentages represent the level of confidence in recognition. So, all other principals are given a  $lcr$  of 0%. We have:

$$OutcomeOfRecognition = \sum_{i=1}^n lcr_i p_i = 0 * p_1 + 0,45 * p_2 + 0,55 * p_3 + \dots + 0 * p_i + \dots + 0 * p_n$$

Technical trust is associated with each vision technique ( $vt$ ): for example face template matching is  $vt1$  with  $tt1$ . Each technique provides a level of recognition ( $lr$ ) for each principal. Assuming that we have  $m$  vision techniques and that each technique is weighted (with  $w$ ) comparatively to other ER schemes used, we have:

$$lcr = \sum_{j=1}^m lr_j * tt_j * w_j$$

If the sum of  $lr$  is too low, this suggests that we need to create a new principal.

Practically, for each different vision recognition technique (face, height and colour), we populated the database as in the previous subsection with 10 persons and then for each remaining sequences (3 or 4 different sequences for each person and 36 sequences in total), we counted how many times each scheme makes the right decision (that is, if the sequence corresponds to person  $i$ , the scheme should recognise



person  $i$ ). We obtained a technical trust of: 0.94 for face template matching (34/36), 0.39 for height matching (14/38) and 0.53 for colour matching (19/36).

## 6 Conclusion

This work demonstrates the applicability of our entity recognition process to computer vision techniques. The use of context and level of confidence in recognition allows us to index and retrieve faster than with a random approach. In fact, the convergence to the true number of people is based not only on image content but also on context. We obtain dynamic enrollment. However, in order to get a database converging to the real number of persons, the system still drops lots of sequences, which are considered of bad quality and could pollute the database.

We have determined that low-cost webcam camera imagery and simple image processing and analysis techniques are sufficient to allow face recognition with a reasonable level of confidence, and other visual attribute recognition with lower, but still useful, levels of confidence. Initial evaluations of the system have yielded promising results and shown that little configuration is needed.

We argue that what is achieved by our system can already be useful for a range of ambient intelligent applications, especially for applications focusing on monitoring rather than security. We speculate that the widespread deployment of our system (which can be done since webcams are widespread and if we release our software) could already raise serious privacy concerns.

## References

- [1] J. Bigun, J. Fierrez-Aguilar, J. Ortega-Garcia, and G.-R. J., "Multimodal Biometric Authentication using Quality Signals in Mobile Communications", in *Proceedings of the 12th International Conference on Image Analysis and Processing*, IEEE, 2003.
- [2] P. Castro and R. Muntz, "Using Context to Assist in Multimedia Object Retrieval", in *ACM Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999, <http://www.info.uqam.ca/~mism/papers/castro.ps>.
- [3] S. Creese, M. Goldsmith, B. Roscoe, and I. Zakiuddin, "Authentication for Pervasive Computing", in *Proceedings of the First International Conference on Security in Pervasive Computing*, 2003.
- [4] J. L. Crowley, J. Coutaz, G. Rey, and P. Reignier, "Perceptual Components for Context Aware Computing", in *Proceedings of Ubicomp'02*, 2002, <http://citeseer.nj.nec.com/541415.html>.
- [5] C. Czirjek, N. O'Connor, S. Marlow, and N. Murphy, "Face Detection and Clustering for Video Indexing Applications", in *Proceedings of Advanced Concepts for Intelligent Vision Systems*, 2003, <http://www.cdv.p.dcu.ie/Papers/ACIVS2003.pdf>.
- [6] A. K. Jain, A. Ross, and S. Prabhakar, "An Introduction to Biometric Recognition", in *IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics*, 2003.
- [7] C. Perez, M. A. Vicente, C. Fernandez, et al., "Aplicacion de los diferentes espacios de color para deteccion y seguimiento de caras.", in *Proceedings of XXIV Jornados de Automatica*, Universidad Miguel Hernandez, 2003.
- [8] J.-M. Seigneur, C. Damsgaard Jensen, S. Farrell, E. Gray, and Y. Chen, "Towards Security Auto-configuration for Smart Appliances", in *Proceedings of the Smart Objects Conference*, 2003, <http://www.grenoble-soc.com/proceedings03/Pdf/45-Seigneur.pdf>.

- [9] J.-M. Seigneur, S. Farrell, C. D. Jensen, E. Gray, and Y. Chen, "End-to-end Trust Starts with Recognition", in *Proceedings of the Conference on Security in Pervasive Computing*, LNCS 2802, Springer, 2003.
- [10] G. Shakhnarovich, L. Lee, and T. Darrell, "Integrated Face and Gait Recognition From Multiple Views", in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2001.
- [11] M. Sonka, V. Hlavac, and R. Boyle, "Image Processing, Analysis, and Machine Vision", Second Edition, PWS Publishing, 1999.
- [12] F. Stajano, "Security for Ubiquitous Computing", ISBN 0470844930, John Wiley & Sons, 2002.
- [13] F. Stajano and R. Anderson, "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks", in *Proceedings of the International Security Protocols Workshop*, 1999, <http://citeseer.nj.nec.com/stajano99resurrecting.html>.
- [14] M. Weiser, "The Computer for the 21st Century", Scientific American, 1991, <http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html>.

## Appendix: Equipment and Software

The COTS low-cost CCD camera with USB interface to a conventional laptop (PentiumIII mobile CPU 866MHz with 256MB RAM) is typical of what is often referred to as a "webcam". It is used in a mode which provides 320x240 pixel 8 bit colour imagery at 15Hz for each channel. Actual resolution and sensitivity are lower due to a colour filter over the CCD and the poor-quality analog-to-digital converter used for quantisation. The camera's focal length is 30mm. Its lens faces the door. The software is written in C++ and uses a MySQL database. The graphical user interface (GUI) is presented in Fig. 3.

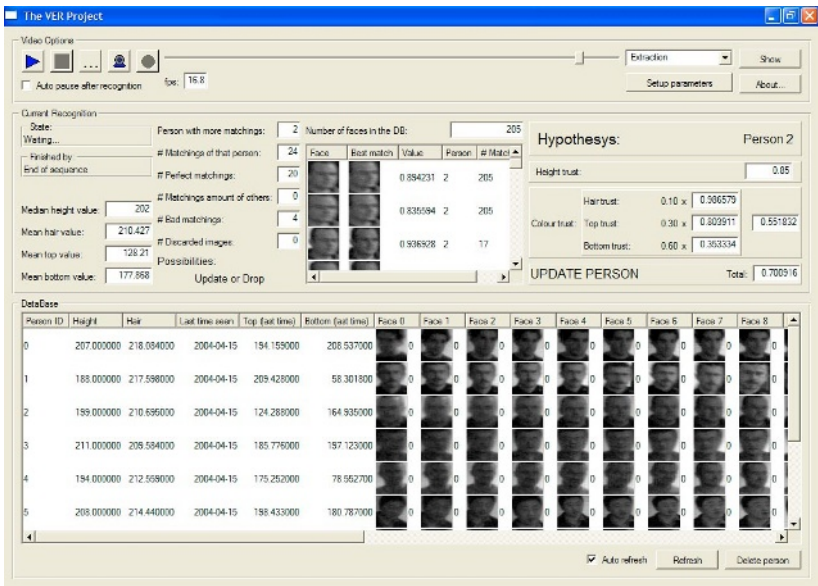


Fig. 3. Software GUI

# A Knowledge Management System for Intelligent Retrieval of Geo-spatial Imagery

Eoin McLoughlin<sup>1</sup>, Dymphna O'Sullivan<sup>1</sup>, Michela Bertolotto<sup>1</sup>, and David Wilson<sup>2</sup>

<sup>1</sup> Smart Media Institute  
Department of Computer Science  
University College Dublin, Dublin, Ireland  
{eoin.A.mcloughlin, dymphna.osullivan,  
michela.bertolotto}@ucd.ie

<sup>2</sup> Software and Information Systems  
University of North Carolina at Charlotte,  
Charlotte, NC 28223, USA  
davils@uncc.edu

**Abstract.** Large repositories of geo-spatial images are employed to support tasks such as intelligence operations, recreational and professional mapping, urban planning and touristic systems. As imagery is retrieved to support a specific task, the interactions, analyses, and conclusions — based on relevant imagery — may be captured together with the images as an encapsulated experience. We are developing annotation-based image retrieval techniques and knowledge-management support for large geo-spatial image repositories incorporating sketch-based querying for image retrieval and manipulation. Leveraging interactive task knowledge to support current user goals requires a smart system that can perform knowledge capture. This paper describes our initial work in intelligent annotation-based image retrieval for geo-spatial imagery systems and presents the task-based knowledge management environment we have developed to support such retrieval.

## 1 Introduction

As a consequence of advances in digital image capture and storage, the geo-sciences, and spatial information engineering, we are experiencing an explosion of available geo-spatial image data. Intelligent systems have become crucial for addressing the problem of imagery information overload by retrieving the most relevant image data and associated information.

From a task-based standpoint, the most relevant work product lies not merely in the applicable visual data, but in descriptions of why and how the information has been collected and how it has been successfully (or unsuccessfully) employed. Capturing and leveraging the human expertise involved in seeking out, distilling, and applying the information required for organisational tasks provides a clear advantage. It serves both to facilitate workflow by providing access to best-practice examples, and to grow a repository of task-based experience as a resource for support, training, and minimizing organisational knowledge-loss as a result of workforce fluctuations.

Our image retrieval system includes a collection of component interfaces for querying a dataset of geo-spatial imagery. A given user may construct a query

outlining their task using a combination of metadata, semantic information, and pen-based sketch input. Our task-based environment enables users to select, compose, and summarize aspects of the digital imagery in support of their current task. From the user's perspective the image retrieval and interaction tools provide them with a convenient way to organize and store a set of insights about relevant imagery by making use of highlighting, transformation, multimedia and annotation tools.

The main purpose of our work is to develop a comprehensive image retrieval system which can extend resulting captured knowledge to make recommendations to other system users as to the best practices to employ in carrying out their own tasks. Both the capture of task knowledge and the subsequent recommendations are carried out unobtrusively using implicit analysis. This involves capturing knowledge at the interface level, maintaining user context based on interactions, selecting the most relevant individual image results based on user context, and effectively presenting encapsulated prior task experiences including annotated imagery. We are particularly interested in developing adaptive content presentation.

This paper presents the current implementation of our image retrieval system. The paper begins with a brief discussion of background and related research. It continues with a description of the image library interaction that provides a baseline for knowledge capture. The paper goes on to describe the user tools available for annotating imagery, capturing experiences and reusing previous rationale based on textual and multimedia annotations. Then it introduces our methods for calculating annotation-based retrieval followed by an initial evaluation of the system. We conclude with a description of future work.

## 2 Background and Related Research

In our research we focus on managing large quantities of geo-spatial information available in raster format, primarily digital aerial photos, satellite images and raster cartography. We have used some pre-existing techniques such as those described in [1] to provide a system that performs efficient geo-spatial database indexing and retrieval. In performing image retrieval we employ a content-based image query mechanism that allows us to locate matching imagery by providing a mapping between relevant imagery and associated metadata information. This is similar to the techniques employed in [2]. Our image retrieval system also provides support for natural user sketch-based interaction. Sketches provide a more intuitive method of communication with a spatial information system as demonstrated by existing systems such as Spatial-Query-By-Sketch [3]. In contrast with the system we are developing, such a system relies on vector data representations and does not include knowledge capture and reuse capabilities.

In this project we are working with large collections of experience and case-based context. Our efforts to capture user context are done automatically. We aim to shield our users from the burden of making explicit queries. Instead we use implicit analysis (e.g., [4, 5]) and observe how they proceed with their task and record this as user context. By situating intelligent tools and support within task environments (e.g., [6]), we can unobtrusively monitor, interpret and respond to user actions concerned with rich task domains based on a relatively constrained task environment model. We

try to predict the users goals and by comparing their current context to that of a previous similar user.

Our methods for annotating multimedia are related to annotating for the semantic web [7] and multimedia indexing [8] where the focus is on developing annotated descriptions of media content. Multimedia database approaches such as QBIC [9] provide for image annotation but use the annotations to contextualise individual images. In this work we are concerned with a more task-centric view of the annotations, where we employ annotations to tell us how an image relates to a task at hand by using image annotations to contextualise task experiences.

3 System Overview

Geo-spatial information represents the location, shape of, and relationships among geographic features and associated artifacts. In this research, we are interested in managing large quantities of geo-spatial information in raster format. As a baseline interaction with the system, we have developed a content-based image query mechanism that can incorporate image metadata information, user task descriptions and user sketches for image retrieval.

When a user logs into the application, they are directed to an interface that enables them to search directly for images that correspond to their current task needs. Queries can be constructed using a combination of metadata, textual task description and pen-based sketch input. The sketch-based query facility is an innovative aspect of our system and the current version does not rely solely on the sketch-based matching algorithms developed in [10]. Our work on integrating the various similarity measures present in the system is ongoing and in this paper we focus on image retrieval using only metadata and task information. For example, in Figure 1, a user might be interested in building an airport near Boston and may enter a query such as the one outlined below.

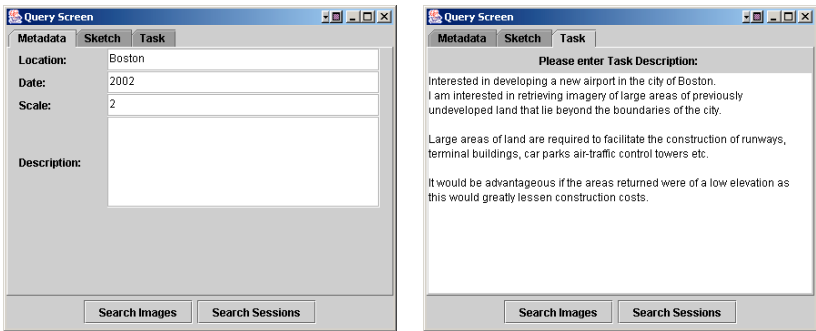
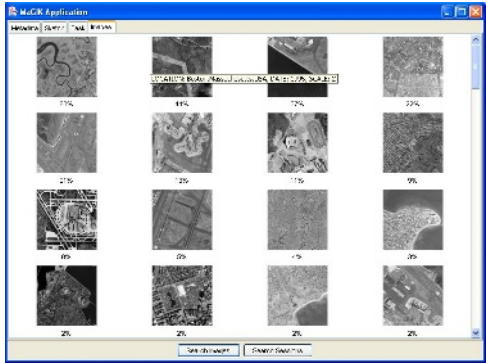


Fig. 1. Query Input Screens

Users may search for images in one of two ways. Firstly they may perform a basic image retrieval search whereby individual images that match their search criteria are returned. Secondly they may search a knowledge base of other users tasks deemed by

the system to be similar to their own context. We refer to the work of each of these individual similar users as “sessions”. These user sessions are constructed by recording all of the contextual knowledge input by a user while addressing their own task goal. The procedure and resulting screens involved in searching the knowledge base for similar sessions are described in the following sections.

If the user chooses to perform a basic image retrieval search the resulting matching images are returned to them as part of the interface in Figure 2. Our methods for performing image retrieval and calculating image matching scores are described in detail in Section 4.1.



**Fig. 2.** Matching Images

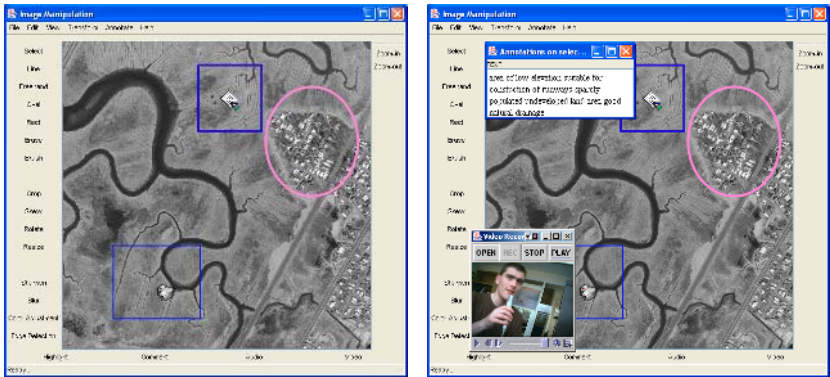
In order to allow multiple candidates to be displayed we present the matching images as a list of thumbnails with an associated matching percentage score. A subset of the most relevant metadata for each image is available as tool tip text when mousing over the image. As the user selects images from the results screen for further analysis the image scores are recalculated automatically and the interface is redrawn to reflect the updated task context. All the selected images are then collected in the current user context and made available for manipulation and annotation.

## 4 Capturing User Context and Task Knowledge for Retrieval

As part of our effort to capture task knowledge to perform image retrieval more effectively by employing the annotations for retrieval we have developed tools for direct image manipulation. These tools include filters, transformations, highlighting, sketching, and post-it type media annotations. For a full description of our image annotation tools please refer to [11]. The tools allow the user to identify regions of interest that can be linked to clarifications and rationale. The manipulations and annotations do not alter the underlying images or geo-spatial information rather they are layered to provide a task-specific view. The tools provide us with a mechanism to capture task context, and thus a basis for computing similarity scores for retrieved images and sessions. This information is collected unobtrusively by the system and is available to the user should they choose to view it.

After retrieving and selecting imagery relevant to the Boston airport, our user can manipulate and/or annotate each image using a substantial set of image annotation tools as shown in Figure 3(a). The user can then go on to add personal media annotations to the image as a whole or to particular highlighted image aspects. Currently the system supports annotation by text, audio and video. The system integrates real-time audio and video capture as well as compression. A wide variety of compression formats are supported, including QuickTime, Mpeg and H.263.

In the case of our airport example the user has annotated the selected image by uploading a textual comment regarding the elevation of the land and how it may be suitable for the construction of an airport (Fig 3(a)). They have recorded and uploaded some media files and highlighted a populated area by sketching a pink oval shape. All textual, audio and video annotations are represented by icons layered on top of the image. All media annotations may also be previewed before being incorporated as part of the knowledge base, and once recorded may be saved as a knowledge parcel associated with the current task.



**Fig. 3.** (a) Image Annotation for Task Context and (b) Viewing Image Annotations

Once a user has annotated an image they may at any time during their session review these annotations by double-clicking and viewing a pop-up description. Also if a user mouses over the icons the region associated with the annotation is emphasized by a dark rectangle drawn around the icon. This functionality is shown above in Fig 3(b).

Capturing task-based knowledge enables a powerful cycle of proactive support. Such a system allows us to facilitate knowledge sharing by retrieving potentially relevant knowledge from other experiences. Currently, we are focusing our annotation-based retrieval on textual annotations. We use information retrieval metrics as a basis for similarity. Given a textual representation of the task context, we can match previously annotated session images to the current context.

The task-based retrieval employs indexes in three separate spaces - Annotation, Image and Session Indexes: For a full description of these indexes please refer to [11].

The indices are used in two different types of retrieval: image retrieval and session retrieval.

## 4.1 Image Retrieval

Task-based image retrieval serves two purposes. First, task-based similarity can be used directly to access annotated images in the image library. Second, it can be integrated with similarities from the other types of query information, such as by image content, to provide a more refined overall metric for retrieval. In searching for relevant images, similarity is computed by first passing the retrieved images through a metadata filter. For the images in the dataset with metadata that match the metadata query entered by the user we compute similarity in the image index. We then compute the average similarities for each attached annotation in the annotation index. Lastly we compute the final image score as the average of overall image and annotation similarities.

## 4.2 Session Retrieval

As the system builds up encapsulated user interactions, another type of retrieval is enabled - retrieving entire previous task-based sessions. This enables a current user to look for the previous image analysis tasks that are most similar to the current task both to find relevant imagery and to examine the decisions and rationale that went into addressing the earlier task.

Figure 4 shows an example of our results for retrieved sessions, displayed in an additional results tab. In order to keep session listings small and still provide enough discriminatory information, each session result is summarized to include the following:

- Percent similarity score
- The most discriminating query or textual task information (if more than one) for the session (since we have captured which results were actually used, we know which queries were most fruitful)
- The most important annotation text (annotations, words, and phrases from the session that have the highest similarity to the current user's context)
- Thumbnail versions of the most important images (those from the session with the highest similarity to the current user's context).

In searching for relevant sessions, similarity is computed by computing similarity in the session index and then for each previous session above a threshold similarity we compute the number of images annotated and browsed in each session as a fraction of the total number of images returned. The final session score is a weighted sum of session similarity and proportion of images annotated and/or browsed.

The proportions of annotated and browsed images provide a measure of the relative usefulness of a given session, and they are given a parameterized weighting relative (currently lower) to the session index similarity component.

If the user wishes to view the annotations made to an image returned in a similar session, they may do so by clicking on the thumbnail, which brings up the image and all its annotations in the image manipulation screen once more (Figure 3(b)). The user may further annotate this image if they wish and/or retain the previous users annotations by adding it to the current session image context.



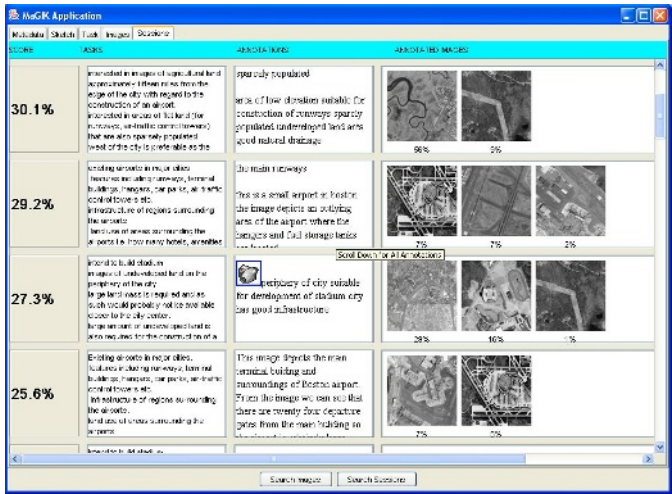


Fig. 4. Similar Sessions

5 Evaluation

For the initial phase of the implementation, we conducted testing with a library of 50 images that were employed to perform 20 different tasks for novice task-domain users. These experiments were designed to test whether the system is performing as expected rather than to provide an absolute measure of utility. Sessions were completed each in one pass, without engaging in feedback and refinement.

5.1 Image Retrieval Evaluation

The goal of our evaluation in this instance was to show that with the addition of annotations, image retrieval improves over time. In order to demonstrate this we performed a relative comparison of the image matching scores of annotated and unannotated images. Firstly an empty library of sessions and annotations was created. Our users interacted with the system to create a series of new sessions. During each session the users added annotations to selected images that they considered relevant to their task. Six different categories of task description were outlined, corresponding to civil development in the following areas: airports, hotels, bridges, hospitals, stadiums and shopping centers. An example task description might be: “build a hospital near the outskirts of the city where infrastructure is good and land is inexpensive.” The users entered task descriptions for each of the outlined categories and initiated searches on the library. The matching scores were recorded for each image, and the user selected images for annotation that seemed relevant to the task description. The annotations for each image were recorded, but not indexed, in order to provide a baseline retrieval performance for the system with annotations. A total of 20 sessions were added. The experiment was then repeated using the same task descriptions in

order to evaluate how the image scores change with the addition of indexed textual annotations.

Figure 5 shows the results for the 4 most relevant images (as judged by the user) for each session. There are six separate queries represented along the X-axis where the light/dark columns (e.g. A, A1) give the similarity respectively without and with indexed annotations for that particular query. The image matching scores were calculated using the similarity metrics discussed in Section 4.1 and are represented on the Y-axis. As expected, there was an overall increase in the image matching scores, demonstrating the usefulness of including textual annotations for retrieval.

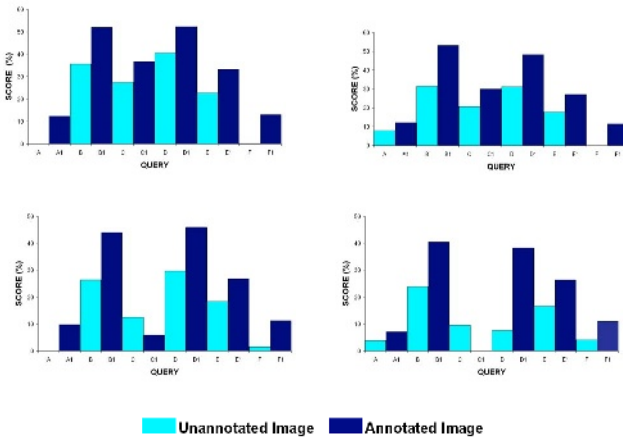


Fig. 5. Image Retrieval Results

## 5.2 Session Retrieval Evaluation

In order to evaluate the retrieval of similar sessions, three new task descriptions (different from the 20 used for evaluating image retrieval) were then constructed in three of the predefined categories. The top three similar sessions returned by the application were analyzed and their results deemed to be relevant or irrelevant.

The first task description corresponded to the “Airport” category. The task description entered by the user outlined that they were interested in viewing images of existing airports as an aid to developing a new airport facility. They sought general information concerning airport locations and orientations with regard to the urban areas that they service, land elevation, infrastructure and the average land space occupied.

The top three similar sessions returned for the airport category were then recorded and analyzed. The scores associated with the top three sessions in this instance were 40.68%, 15.14% and 14.82% respectively. The task description of the first similar session outlined a scenario where the user was interested in constructing a new airport. It differed from the task description of the current session, however, in that the user was not interested in retrieving images of existing airports. Rather, they simply wished to view areas of land that would be appropriate for such a new

development. Both task descriptions contained text associated with the airport domain such as elevation and land-space, as did some of the annotations uploaded by the user of the similar session to the images returned in their session. This session was deemed to be useful in fulfilling our current task description.

In the second similar session the user had entered a task requesting the retrieval of images in order to analyze general land usage in the selected cities. Some of the images returned by the similar session depicted airports, and the user had made annotations in this regard. The session score was higher because some more general use terminology, such as land and urban appear in both queries. The session was deemed moderately related to our current task.

The third most similar session user was interested in constructing a stadium in an urban area. The tasks are similar in that the land sites required for both developments are relatively large when compared to many other development domains and this is reflected in the session score. In both cases it is preferable for the development sites to be located away from the center of urban areas, given cost and the scale of the development. This session also seems to see gains in similarity from more general use terminology that parallels the domains. We expect that there would be more marked differences allowing for finer distinctions with task-domain experts and more substantial task descriptions. This session was deemed to be quite similar to ours, but not very relevant.

Similar evaluations were carried out in the development categories of “Bridge”—top 3 sessions: 35.59% (relevant), 14.65% (very relevant), and 13.79% (not relevant) — and “Hotel”—top 2 sessions: 52.19% (relevant) and 20.14% (relevant). Only 2 sessions were retrieved in total for the “Hotel” task. While these early results are only indicative, they do show that the system is performing as expected. We intend to undertake larger scale testing in the near future and realize that there are many factors that will need to be accounted, including a larger range of categories, scaling the number of annotations, and refining vocabulary toward more domain-specific usage.

## 6 Conclusions and Future Work

We have introduced our approach to developing intelligent knowledge-based image retrieval for geo-spatial applications. We are currently carrying out more extensive testing of the system using precision and recall metrics to test our retrieval algorithms. We also hope to conduct trials with domain experts in the future. We are in the process of transferring the system to the mobile platform and we are also investigating the possibility of adding other resources such as automatic speech recognition (ASR) in order to facilitate media retrieval. As part of our research we are investigating the possibilities for exposing aspects of captured knowledge in the interface, such as finer visual cues in the cues in iconography and maintaining orientation as a result of presentation changes in response to user context. Finally we note that our system has wider potential applications beyond the geo-spatial context, such as medical imagery, and we expect that these techniques will prove valuable in many fields that rely on image analysis.

**Acknowledgements.** The support of the Research Innovation Fund initiative of Enterprise Ireland is gratefully acknowledged.

## References

1. Shekhar, S., Chawla, S., Ravada, S., Fetterer, A., Liu, X., and Lu, C. Spatial databases: Accomplishments and research needs. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 11(1):45–55, 1999.
2. Frew, J., Freeston, M., Hill, L., Janeé, G., Larsgaard, M., & Zheng, Q. (1999). Generic query metadata for geo-spatial digital libraries. *Proceedings of the Third IEEE Meta-Data Conference (Meta-Data '99)*, April 6-7, 1999, Bethesda, MD, sponsored by IEEE, NOAA, Raytheon ITSS Corp., and NIMA.
3. Egenhofer, M. J. Spatial-query-by-sketch. In M. Burnett and W. Citrin eds, editors, VL'96: IEEE Symposium on Visual Languages, Boulder, CO, pages 60–67, 1996.
4. Claypool, M., Brown, D., Le, P. and Wased, M. Implicit interest indicators. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 33–40, 2001.
5. O'Sullivan, D., Smyth, B. and Wilson, D. Explicit vs. implicit profiling— a case-study in electronic programme guides. In *Proceedings of the 18<sup>th</sup> International Joint Conference on Artificial Intelligence (IJCAI-03)*, Acapulco, Mexico 2003.
6. Budzik, J., Birnbaum, L., and Hammond, K. Information access in context. *Knowledge Based Systems*, 14(1-2):37–53, 2001
7. A. Champin, Y.Prié, and A. Mille. Annotating with uses: a promising way to the semantic web. In *K-CAP 2001 Workshop on Knowledge Markup and Semantic Annotation*, pages 79-86, 2001.
8. Worring, M., Bagdanov, A., Gemerr, J., Geusebroek, J., Hoang, M., Schrieber, A. T., Snoek, C., Vendrig, J., Wielemaker, J. and Smeulders, A. Interactive indexing and retrieval of multimedia content. 2002.
9. Flickner, M., Sawhney, H., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D. and Yanker, P. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23–32, 1995.
10. Carswell, J.D., 2000. Using Raster Sketches for Digital Image Retrieval: Ph.D. Thesis, Department of Spatial Information Science and Engineering, The University of Maine.
11. Wilson, D.C., Bertolotto, M., McLoughlin, E., O'Sullivan, D. Knowledge Capture and Reuse for Geo-Spatial Imagery Tasks. In *Proceedings of the 5<sup>th</sup> International Conference on Case-Based Reasoning (ICCBR-03)*, pages 622-636, Trondheim, Norway, 2003.

# An Adaptive Image Content Representation and Segmentation Approach to Automatic Image Annotation

Rui Shi<sup>1</sup>, Huamin Feng<sup>1,2</sup>, Tat-Seng Chua<sup>1</sup>, and Chin-Hui Lee<sup>3</sup>

<sup>1</sup> School of Computing, National University of Singapore, Singapore  
{shirui, fenghm, chuats}@comp.nus.edu.sg

<sup>2</sup> Beijing Electronic Science & Technology Institute, 100070, China

<sup>3</sup> School of Electrical and Computer Engineering, Georgia Institute of Technology,  
Atlanta, GA, USA  
chl@ece.gatech.edu

**Abstract.** Automatic image annotation has been intensively studied for content-based image retrieval recently. In this paper, we propose a novel approach to automatic image annotation based on two key components: (a) an adaptive visual feature representation of image contents based on matching pursuit algorithms; and (b) an adaptive two-level segmentation method. They are used to address the important issues of segmenting images into meaningful units, and representing the contents of each unit with discriminative visual features. Using a set of about 800 training and testing images, we compare these techniques in image retrieval against other popular segmentation schemes, and traditional non-adaptive feature representation methods. Our preliminary results indicate that the proposed approach outperforms other competing systems based on the popular Blobworld segmentation scheme and other prevailing feature representation methods, such as DCT and wavelets. In particular, our system achieves an  $F_1$  measure of over 50% for the image annotation task.

## 1 Introduction

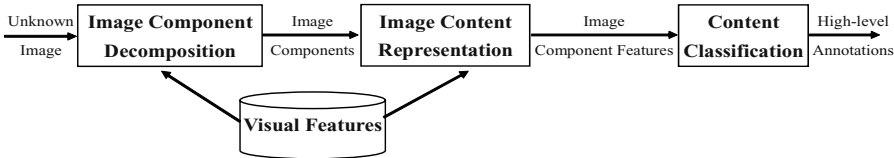
Recent advances in computer, telecommunications and consumer electronics have brought forward a huge amount of images to a rapidly growing group of users. With the wide spread use of Internet, more and more digital images are now available on the World-Wide Web (WWW). Thus effective tools to automatically index images are essential in order to support applications in image retrieval. In particular, image annotation has become a hot topic to facilitate content-based indexing of images.

Image annotation refers to the process of automatically labeling the image contents with a predefined set of keywords representing image semantics. It is used primarily for image database management. Annotated images can be retrieved using keyword-based search, while non-annotated images can only be found using image-based analysis techniques, which are still not very accurate nor robust. Thus automatic image annotation (AIA) aims to invest a large amount of preprocessing efforts to annotate the images as accurately as possible to support keyword-based image search.

Recent studies [14] suggest that users are likely to find it more useful and convenient to search for images based on text annotations rather than using visual-based features.

Most current automatic image annotation (AIA) systems are composed of three key modules: image component decomposition, image content representation, and content classification. A general framework of AIA is shown in Figure 1. The image component decomposition module decomposes an image into a collection of sub-units, which could be segmented regions, equal-size blocks or the entire image. The image content representation module models each content unit based on a feature representation scheme. Finally, the image content classification module computes the association between unit representations and textual concepts and assigns appropriate high-level concepts to the sub-image. Hence we need to answer the following questions:

- 1) What image components should be used as image analysis units?
- 2) How to develop better feature representation to model image contents?
- 3) How to describe the relationships between image components so as to build the relationships between these components and high-level annotations?



**Fig. 1.** The general framework of Automatic Image Annotation (AIA) system

To address these three problems, most recent research focused on Problems 1 and 3. For Problem 1, three kinds of image components are often used as image analysis units in most CBIR (content-based image retrieval) and AIA systems. In [11, 18], the entire image was used as a unit. Only global features can be used to represent images. Such systems are usually not effective since the global features cannot capture the local properties of an image well. Some recent systems use segmented regions as sub-units in images [3, 5]. However, the accuracy of segmentation is still an open problem. To some extent, the performance of annotation or retrieval depends on the results of segmentation. As a compromise, several systems adopt fixed-size sub-image blocks as sub-units for an image [12, 19]. The main advantage is that block-based methods can be implemented easily. In order to compensate for the drawbacks of block-based method, hierarchical multi-resolution structure is employed [23]. When compare with region-based methods, the block-based methods often result in worse performance.

A lot of research work has also been conducted to tackle Problem 3. In [23], 2-D MHHMs are used to characterize the relationship between sub-image blocks. This model explores statistical dependency among image blocks across multiple resolution

levels as well as within a single resolution. In [17], the relationship between image regions with spatial orderings of regions is considered by using composite region templates. In [8], cross-media relevance models are used to describe the association between segmented regions and keywords.

As far as we know, not much research has been done to address Problem 2. Conventional CBIR approaches employ color, simple texture, and statistical shape features to model image content [11, 16, 18]. It is well known that such low-level content features are inadequate, and thus the resulting CBIR system generally has low retrieval effectiveness. Moreover, the retrieval effectiveness depends largely on the choice of query images, and the diversity of relevant images in the database. In order to ensure high accuracy, special purpose systems tend to rely on domain-specific features, such as the use of face detectors and face recognizers to look for images of people [11]. Since a single, fixed content representation is unlikely to meet all the needs of different applications, the challenge here is to explore adaptive content representation schemes to support a wide range of classification tasks.

In this paper we propose an adaptive and effective representation that has a high content representational power beyond the traditional low-level features, such as color, texture and shapes. In particular, we extend the adaptive matching pursuit (MP) features [2, 10] to model the texture content of images. In conjunction with these adaptive features, we also propose a two-level segmentation method to partition the image content into more appropriate units. We adopt the SVM-based classifiers employed in [7] to perform image annotation. We evaluated the proposed framework on an image annotation task using a set of about 800 training and testing images selected from the CorelCD and PhotoCD image collections. Preliminary results showed that the adaptive approach outperformed other competing systems based on the popular Blobworld segmentation scheme and other prevailing feature representation methods, such as DCT and wavelets. In particular, our system achieves an  $F_1$  measure of over 50% on the image annotation task.

The rest of the paper is organized as follows. In Section 2 we introduce the proposed framework with adaptive MP features and two-level segmentation. In Section 3 we discuss experimental results and compare our performance with other systems. Finally we conclude our findings in Section 4.

## 2 Adaptive Image Content Representation

As discussed earlier, almost all the existing systems used a combination of color, texture and statistical shape features to model the visual contents of images [11, 16, 18]. These features have been found to be too low-level to adequately model the image content. Because the discrimination power of these visual features is usually quite low, they are effective only in matching highly similar images, and often fail if there exist diversity in relevant images, or when the query is looking for object segments within the images. These problems point to the need to develop adaptive scheme, in which feature representation can be adapted to suit the characteristics of the images to be modeled. Here we propose the adaptive texture features based on

matching pursuit (MP) [2, 10], to be used in conjunction with color histogram, to represent the image content. We do not use the shape feature as it is often unreliable and easily affected by noise. As a part of this work, we also propose a two-level segmentation method to segment the image content into meaningful units. In the following, we introduce the proposed adaptive MP features and the two-level segmentation method.

## 2.1 Adaptive Texture Features Based on Matching Pursuit

Tuceryan and Jain [20] identified five major categories of features for texture identification: statistical, geometrical, structural, model-based, and signal processing features. In particular, signal processing features, such as DCT, wavelets and Gabor filters, have been used effectively for texture analysis in many image retrieval systems [11, 15, 17]. The main advantage of signal processing features is that they can characterize the local properties of an image very well in different frequency bands. However, specific images usually contain a lot of local properties that need to be characterized individually. In order to facilitate adaptive image representation, we borrow the concept from matching pursuit [2, 10] and employ a combination of DCT and three wavelets as the basis functions to construct an over-complete dictionary in our system. The three wavelets chosen are Haar, Daubechies and Battle/Lemarie, which are often used for texture analysis [20, 23]. We did not use the Gabor filters because they are non-orthogonal and expensive to construct.

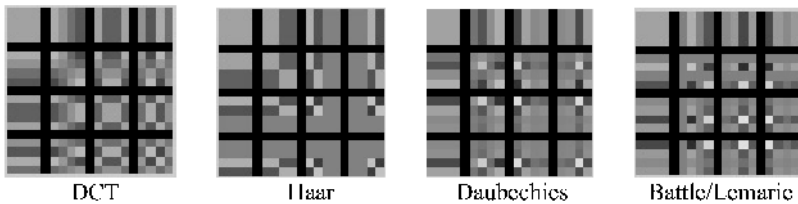


Fig. 2. All the basis functions employed in our system

The basis functions for DCT and different types of wavelets are shown in Figure 2. It can be seen that these basis functions have different abilities to represent the details of images using different local properties. For example, images with sharp edges such as modern buildings are better modeled using Haar wavelets; signals with sharp spikes are better analyzed by Daubechies' wavelets; whereas images with soft edges such as clouds are better modeled using DCT basis functions. Thus given a band of basis functions for DCT and different wavelets, we should be able to find a representation that best matches the image content.

The basic idea behind wavelet and DCT transforms is similar. In DCT, a signal is decomposed into a number of cosines of different frequency bands; whereas in wavelet transform, a signal is decomposed into a number of chosen basis functions. To extract the MP features, we divide the image into fixed size blocks of 4x4 pixels.



All these basis functions are then partitioned into 16 frequency bands in the horizontal, vertical and diagonal directions of DCT and wavelet transforms.

The algorithm for adaptive MP texture feature extraction can be described as follows. Let  $F=f(x,y)$ , ( $1 \leq x \leq N$ ,  $1 \leq y \leq N$ ,  $N=4$ ), where  $F$  is an 2D image block and  $f(x,y)$  denotes the intensity value at location  $(x,y)$ . We first transform the 2D image block  $F$  into a vector  $I$ , column by column or row by row, with  $I=(f(1,1), f(1,2), \dots, f(N,N))^T$ ,  $N=4$ . Thus, there are a total of  $N \times N$  elements in vector  $I$ . We construct all the basis functions in the over-complete dictionary [2, 10] in a similar manner.

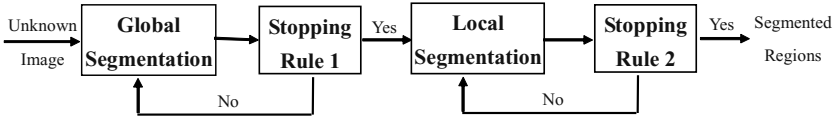
Next, we assumed that the signal space  $\Omega$  is an  $R^{N \times N}$  Hilbert space, with an inner product  $\langle \cdot, \cdot \rangle$ , and an induced norm denoted as  $\|I\|=\langle I, I \rangle^{1/2}$ . Assume that  $D \subset \Omega$  is a dictionary of  $M$  basis functions in  $\Omega$ , with  $D=\{w_1, w_2, \dots, w_M\}$ . Without loss of generality, it is assumed that  $\|w_j\|=1$  for every basis function (or word),  $w_j$ . Then, the feature extraction algorithm can be described in the following steps:

- 1) Set  $I_0 = I$ ,  $j=1$ .
- 2) Compute  $w_j = \arg \max_{w \in D} \langle I_{j-1}, w \rangle^2$ .
- 3) Let  $a_j = \langle I_{j-1}, w_j \rangle$  and  $I_j = I_{j-1} - a_j w_j$ .
- 4) Repeat Steps 2 and 3 with  $j \leftarrow j+1$  until  $\|I_j\|^2 < \epsilon$ , a pre-fixed threshold.
- 5) Compute the energy value for each frequency band by the coefficient vector  $(a_1, a_2, \dots, a_M)$  which is obtained in Steps 1-4. Thus the 16-dimension vector of energy value is used as adaptive MP texture feature for each image block.

Comparing with the conventional wavelet-based texture features, the main advantages of our adaptive MP texture features are that they are efficient and provide adaptive reflection of local texture properties. This is because we are able to obtain the most appropriate representation for an image with the fewest significant coefficients through matching pursuit.

## 2.2 An Adaptive Two-Level Image Segmentation Method

In [22], it is argued that an internal spatial/frequency representation in human vision system is capable of preserving both global information and local details. In order to emulate human vision perception, we propose a two-level segmentation method to segment an image into meaningful regions by taking advantage of the adaptive MP features introduced above. Compared with the traditional multi-resolution and hierarchical segmentation methods [5, 6], our algorithm does not need to build complex segmentation schemes, including hierarchy structure for organizing and analyzing image content, and criteria for growing and merging regions. In addition, as the segmentation problem is task dependent, we consider segmentations at different levels as different tasks. Since the segmentation from global level to local level is a process of gradual refinement, we therefore use different features and methods for segmentations at different levels, as shown in Figure 3.



**Fig. 3.** The process of our two-level segmentation method

The image is first partitioned into a large number of small 4x4 blocks. At the global level, we extract mostly global features such as color, texture and position for each block. Here, color is a 3-dim feature vector depicting the average (L, u, v) color components in a block. We apply DCT transform to the L component of the image block and extract a 3-dim texture vector, consisting of the energy values of the frequency bands in the horizontal, vertical and diagonal directions. Finally, we append the center position of the block (x,y) to the feature vector. We perform global segmentation using GMM (Gaussian Mixture Model), which has been used extensively and successfully in automatic speech and speaker recognition [9] to model non-Gaussian speech features. In order to reduce the complexity in estimating the GMM parameters, we adopt a diagonal covariance matrix with the same variance for all elements. In the meantime, we employ the MDL principle [13] to determine the number of mixture components, and use it as the Stopping Rule 1 in Figure 3. The number of mixture components used ranges from 2 to 4.

For local segmentation, we use employ the adaptive MP features as discussed in Section 2.1 as it better model the content of global regions. For the 19-dim MP feature vector, three are the average of (L, u, v) color components in a 4x4 block, and the other 16 are the adaptive MP texture features. We adopt the K-means algorithm to perform local segmentation. Since the K-means algorithm does not specify the number of clusters, we adaptively choose  $k$  by gradually increasing its value ( $k=1, 2$  or  $3$ ) until the distortion  $D(k) - D(k-1)$  is below a threshold, with

$$D(k) = \sum_{i=1}^N \min_{1 \leq j \leq k} (x_i - \hat{x}_j)^2 \quad (1)$$

where  $\{x_i : i=1, \dots, N\}$  are the  $N$  observations, and  $\{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_k\}$  are the  $k$  group means. Eq. (1) also serves as the basis for Stopping Rule 2 in Figure 3.

Figure 4 compares two segmentation results obtained using the proposed two-level scheme and Blobworld. It can be seen that our segmentation method tends not to over-segment the regions as in the case for Blobworld. There are two possible reasons. First, we need to estimate fewer parameters. The estimates tend to be more accurate with less training sample than those based on the full covariance matrix used in the Blobworld algorithm. Second, our method segments an image into regions in two steps by considering both global and local features, thus it can reduce over-segmentation.

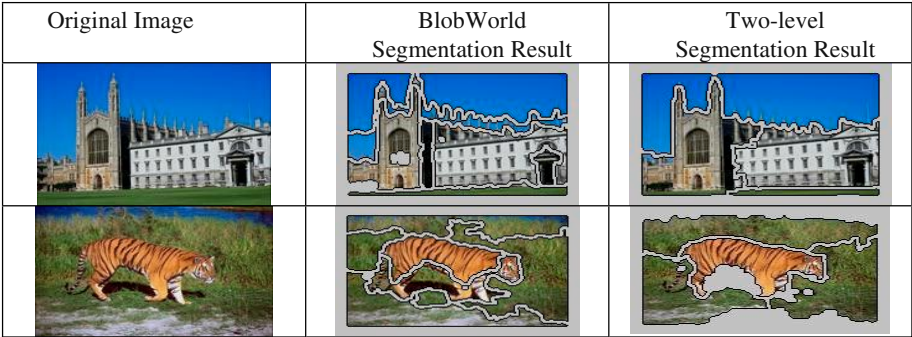


Fig. 4. Some segmentation results by Blobworld and our two-level segmentation method

3 Testing and Results

In order to evaluate the discrimination power of the new adaptive features and the effectiveness of our two-level segmentation method, we designed two sets of experiments based on about 800 images, chosen from the Corel and PhotoCD image collections. These images were first segmented into regions by using Blobworld and our two-level segmentation method. For our method, we used the second (or local) level of segmentation as the basis for annotation. After segmentation, we obtained over 4,000 segmented regions by Blobworld, and over 5,000 second level regions by our two-level segmentation method. To prepare the ground truth, we manually tagged each resulting segmented region using one of 23 concepts as listed in Table 1. The concepts are chosen based on the hierarchical concepts described in TGM I (Thesaurus for Graphics Materials) [24]. We chose these concepts based on the following two criteria: (a) the concept has concrete visual signatures; and (b) we are able to gather sufficient number of images for training and testing. When manually tagging the segmented regions, we came across many fragmented and meaningless regions because of the problems of segmentation methods. For such regions, we simply tagged them as „none“. During testing, when „none“ is detected, we simply discard it.

We employed SVM with RBF kernels [4] to perform image and sub-image classification as is done in [7]. In the training stage, we train a binary SVM model for each concept. In the testing stage, we pass the un-annotated segmentation of an image through all the models, and assign only the concept that corresponds to the model giving the highest positive result to the segment. Thus the training is based on segmented regions; while testing is done at the image level. We performed 10 fold cross validation by randomly choosing 80% of images from the corpus for training, and the remaining 20% for testing.

Table 1. The list of 23 concepts used to annotate images.

animals, vehicles, beaches, mountains, meadows, buildings, transportation facilities, office equipments, food, clouds, sky, snow, sunrises/sunsets, grasses, trees, plants, flowers, rocks, clothing, people, water, none, unknown
--

Experiment 1 aims to evaluate the effectiveness of our adaptive MP features, independent of the segmentation method used. We therefore used only the segmentations produced by the popular Blobworld method to perform image annotation. We tested the performance of the systems by combining color histogram with different texture models, including DCT, Haar wavelet, Daubechies' wavelet, Battle/Lemarie wavelet, and our adaptive MP features, as shown in Table 2. The top four rows of Table 2 provide baseline benchmarks. It can be seen that our MP features produced the best image annotation results in terms of both recall and  $F_1$  measures. The precision results are slightly worse than those obtained with the other competing features. This indicates that the discrimination power of the adaptive MP features can still be enhanced with an adaptive segmentation algorithm.

Experiment 2 was designed to test the effectiveness of the adaptive MP features using the proposed two-level segmentation method. We used a combination of global feature, color histogram and different texture features (same as in Experiment 1) to model the content. The global features are available only in our two-level segmentation method (see Section 2.2); but not in Blobworld method. The global feature used here is a 6-dim vector, consisting of 3 for Luv mean and 3 for DCT textures. The results of Experiment 2 are listed in Table 3, which again shows that the use of adaptive MP texture features give the best image annotation performance as compared to all other feature combinations. In fact, it produces the best  $F_1$  measure of over 0.5 when using both two-level segmentation and adaptive MP feature extraction.

Tables 2 and 3 also demonstrate that the proposed two-level segmentation scheme is clearly superior to the single-level Blobworld method, because the results in Table 3 are consistently better than those in Table 2 in all cases, especially for recall and  $F_1$ .

**Table 2.** Results of Experiment 1 based on Blobworld (single-level) segmentation.

Feature Type	Recall	Precision	$F_1$
Luv hist + [DCT]	0.2859	0.3867	0.3287
Luv hist + [Haar]	0.2907	0.3972	0.3357
Luv hist + [Daube]	0.2791	0.3910	0.3257
Luv hist + [Battle]	0.2901	0.3920	0.3334
Luv hist + MP features	0.3544	0.3836	0.3684

**Table 3.** Results of Experiment 2 based on our two-level segmentation.

Feature Type	Recall	Precision	$F_1$
Global features+Luv hist+[DCT]	0.512	0.4022	0.4505
Global features+Luv hist+[Haar]	0.5181	0.4079	0.4564
Global features+Luv hist+[Daube]	0.5124	0.4104	0.4558
Global features+Luv hist+[Battle]	0.519	0.4132	0.4601
Global features+Luv hist + MP features	0.5642	0.454	0.5031

## 4 Conclusion

In this paper, we proposed a novel adaptive content representation scheme with two key components: (a) adaptive matching pursuit feature extraction for texture; and (b) adaptive two-level segmentation method. We compared our proposed methods with popular single-level segmentation, like Blobworld, and conventional feature representations, based on color and DCT or wavelets. The results indicate that our proposed adaptive approach outperformed other competing methods in most combinations, especially when the two-level segmentation algorithm is employed. In particular, our combined scheme achieved an  $F_1$  measure of over 50%. The overall results suggest that some of the difficulties in content-based image retrieval and automatic image annotation could be mitigated by jointly considering both the segmentation and feature representation issues. It will also be worth looking into simultaneous segmentation and classification, commonly done in the state-of-the-art automatic speech and speaker recognition systems.

Our future research is focused in two directions. First, we will refine our adaptive approach and test it on a large collection of image sets. Second, we will extend the proposed techniques to handle video and web-based multimedia contents.

## References

- [1] Y. A. Aslandogan and C. T. Yu, „Multiple evidence combination in image retrieval: Dohenese searches for people on the web,“ *ACM SIGIR'2000*, Athens, Greece, 2000.
- [2] F. Bergeaud and S. Mallat, „Matching pursuits of images,“ *Proc. IEEE ICIP'95*, Vol. 1, pp. 53-56, Washington DC, Oct 1995.
- [3] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein and J. Malik, „Blobworld: A system for region-based image indexing and retrieval,“ *Proc. Int'l Conf. Visual Information System*, 1999.
- [4] C.-C. Chang and C.-J. Lin, „Training nu-support vector classifiers: theory and algorithms,“ *Neural Computation*, 13 (9), pp. 2119-2147, 2001.
- [5] Y. Deng, B. S. Manjunath, and H. Shin, „Color image segmentation,“ in *Proc. IEEE CVPR*, 1999.
- [6] P. Duygulu and F. Y. Vural, „Multi-Level image segmentation and object representation for content based image retrieval,“ *SPIE Electronic Imaging 2001, Storage and Retrieval for Media Databases*, January 21-26, 2001, San Jose, CA.
- [7] H.M. Feng and T.S. Chua, „A Bootstrapping Approach to Annotating Large Image Collection“. ACM SIGMM International Workshop on Multimedia Information Retrieval. Berkeley, Nov 2003. 55-62.
- [8] J. Jeon, V. Lavrenko and R. Manmatha, „Automatic image annotation and retrieval using cross-media relevance models,“ *ACM SIGIR'03*, July 28-Aug 1, 2003.
- [9] C.-H. Lee, F. K. Soong and K. K. Paliwal, *Automatic Speech and Speaker Recognition: Advanced Topics*, Kluwer Academic Press, 1996.
- [10] S.G. Mallat and Z.F. Zhang, „Matching pursuits with time-frequency dictionaries,“ *IEEE Trans. on Signal Processing*, 41 (12), pp. 3397-3415, 1993.

- [11] B. Manjunath and W. Ma, „Texture features for browsing and retrieval of image data,“ *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837-842, Aug 1996.
- [12] Y. Mori, H. Takahashi and R. Oka, „Image-to-word transformation based on dividing and vector quantizing images with words,“ In *Proc. of First International Journal of Computer Vision*, 40(2): 99-121, 2000.
- [13] J. Rissanen, „Modeling by shortest data description,“ *Automatica*, 14:465-471, 1978.
- [14] K. Rodden, „How do people organize their photographs?“ In *BCS IRSG 21<sup>st</sup> Ann. Colloq. on Info. Retrieval Research*, 1999.
- [15] M. Shenier and M. Abedel-Mottaleb, „Exploiting the JPEG compression scheme for image retrieval,“ *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18 (8), 849-853, 1996.
- [16] J.R. Smith and S.F. Chang, „VisualSeek: A fully automated content-based query system,“ *ACM Multimedia*, 1996.
- [17] J.R. Smith and C.S. Li, „Image classification and querying using composite region templates,“ *Journal of Computer Vision and Image Understanding*, 2000.
- [18] M. Swain and D. Ballard, „Color indexing,“ *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.
- [19] M. Szummer and R.W. Picard, „Indoor-outdoor image classification,“ *IEEE Intl Workshop on Content-based Access of Image and Video Databases*, Jan 1998.
- [20] M. Tuceryan and A.K. Jain, „Texture analysis,“ *Handbook Pattern Recognition and Computer Vision*, Chapter 2, pp. 235-276, World Scientific, 1993.
- [21] M. Unser, „Texture classification and segmentation using wavelet frames,“ *IEEE Trans. on Image Processing*, 4 (11), pp. 1549-1560, 1995.
- [22] R. D. Valois and K.D. Valois, „Spatial Vision,“ New York: Oxford, 1988.
- [23] J.Z. Wang and J. Li, „Learning-based linguistic indexing of pictures with 2-D MHMMs,“ *Proc. ACM Multimedia*, pp. 436-445, Juan Les Pins, France, Dec 2002.
- [24] <http://www.loc.gov/rr/print/tgm1/>

# Knowledge Assisted Analysis and Categorization for Semantic Video Retrieval\*

Manolis Wallace, Thanos Athanasiadis, and Yannis Avrithis

Image, Video and Multimedia Systems Laboratory  
School of Electrical and Computer Engineering  
National Technical University of Athens  
9, Iroon Polytechniou St., 157 73 Zographou, Greece  
{wallace, thanos, iavr}@image.ntua.gr

**Abstract.** In this paper we discuss the use of knowledge for the analysis and semantic retrieval of video. We follow a fuzzy relational approach to knowledge representation, based on which we define and extract the context of either a multimedia document or a user query. During indexing, the context of the document is utilized for the detection of objects and for automatic thematic categorization. During retrieval, the context of the query is used to clarify the exact meaning of the query terms and to meaningfully guide the process of query expansion and index matching. Indexing and retrieval tools have been implemented to demonstrate the proposed techniques and results are presented using video from audiovisual archives.

## 1 Introduction

The advances in multimedia databases and data networks along with the success of standardization efforts of MPEG-4 [1] and MPEG-7 [2] have driven audiovisual archives towards the conversion of their manually indexed material to digital, network accessible resources, including video, audio and still images. By the end of last decade the question was not on whether digital archives are technically and economically viable, rather on how they would be *efficient* and *informative* [3]. In this framework, different scientific fields, such as database management, image/video analysis, computational intelligence and the semantic web, have observed a close cooperation [4].

Access, indexing and retrieval of image and video content have been dealt either with content-based, or metadata-based techniques. In the former case, image and video content is analyzed, visual descriptors are extracted and content-based indexes are generated, to be used in query by example scenarios [5]. It has been made clear however, that query by visual example is not able to satisfy multiple search usage requirements [6]. In the latter case, various types of metadata, mainly textual, are typically attached to the original data and used to match against user queries. Although this makes textual (e.g., keyword) search possible, to which users are more accustomed, the main disadvantage of this approach is the lack of semantic interpretation of the queries that may be posed [7]

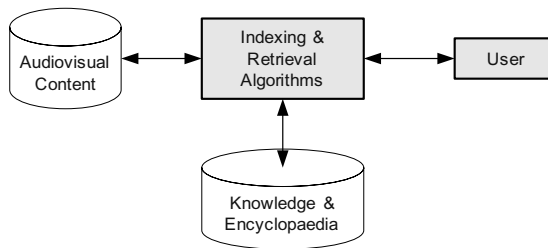
---

\* This work has been partially funded by the EU IST-1999-20502 project FAETHON.

The proposed technique achieves semantic handling of archive content using an encyclopedia which contains definitions of abstract semantic classes. The creation of the encyclopedia relies both on human experts and existing ontologies. During document analysis and indexing, semantic entities of the multimedia document descriptions are linked to the abstract ones of the encyclopedia. During retrieval, the supplied keywords of the user query are translated into the semantic entities and the documents whose descriptions have been linked to the requested semantic entities are retrieved. Fuzzy relations are used for knowledge representation, and fuzzy algebraic techniques for query analysis, video document indexing and matching between the two. After providing the general structure of the proposed video retrieval approach, the paper presents the proposed knowledge representation and continues by discussing the application of this knowledge in analysis and retrieval. Indicative results are provided on indexing and retrieval of content from audiovisual archives.

## 2 Overview

Three main entities participate in the process of video retrieval are: the user, the actual video content and the knowledge that is available to the system, as depicted in Fig. 1. Since each one is expressed in a fundamentally different way, the main effort is to produce techniques that can achieve a uniform representation of all three, so that information provided from each one may be combined and matched. In this work, uniform representation is attempted at a semantic level.



**Fig. 1.** The proposed framework for knowledge-assisted video indexing and retrieval.

The encyclopedia contains definitions of both simple objects and abstract object classes. Simple objects can be automatically detected in a video document by matching either their visual and audio descriptors with the corresponding ones in the encyclopedia, stored using the structures of Fig 2., or the metadata descriptions with the textual descriptions of the objects. Visual descriptors are extracted using algorithms described in [8], and linked to object/event definitions in the encyclopedia. Abstract classes and concepts cannot be automatically detected in the video documents; they have to be inferred from the simple objects that are identified, thus semantically enriching the indexing of the documents. The user query is issued in a textual form. It is then automatically mapped to semantic entities found in the encyclopedia, and expanded so as to include entities that are not mentioned but implied. The query in its



semantic form is then matched to the semantic indexing of the document, thus providing the system response. This response, being constructed in a semantic way, is much more intuitive than the one provided by existing video retrieval systems. In the following sections we briefly describe the knowledge representation utilized in this work, and explain how it is applied in offline and online tasks.

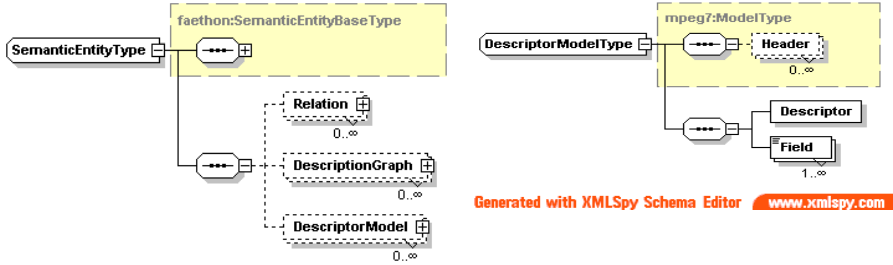


Fig. 2. The structures used to represent the visual and audio characteristics of semantic entities.

### 3 Knowledge Representation and Context Detection

#### 3.1 Fuzzy Relational Knowledge Representation

Although any type of relation may be contained in an ontology, the two main categories are taxonomic (i.e. ordering) and compatibility (i.e. symmetric) relations. Compatibility relations have traditionally been exploited by information retrieval systems for tasks such as query expansion. They are ideal for the description of similarities, but fail to assist in the determination of context; the use of ordering relations is necessary for such tasks. Thus, a challenge of intelligent information retrieval is the meaningful exploitation of information contained in taxonomic relations of an ontology.

The specialization relation  $Sp$  is a fuzzy partial ordering on the set of semantic entities.  $Sp(a,b) > 0$  means that the meaning of  $a$  includes the meaning of  $b$ . The context relation  $Ct$  is also a fuzzy partial ordering on the set of semantic entities.  $Ct(a,b) > 0$  means that  $b$  provides the context for  $a$  or, in other words, that  $b$  is the thematic category that  $a$  belongs to. Other relations considered in the following have similar interpretations. Their names and corresponding notations are given in Table 1.

Fuzziness of the aforementioned relations has the following meaning: high values of  $Sp(a,b)$  imply that the meaning of  $b$  approaches the meaning of  $a$ , while as  $Sp(a,b)$  decreases, the meaning of  $b$  becomes narrower than the meaning of  $a$ . A similar meaning is given to fuzziness of other semantic relations as well. The knowledge contained in these relations is combined into a single quasi-taxonomic relation as follows [9]:

$$T = (Sp \cup Ct^{-1} \cup Ins \cup P \cup Pat \cup Loc \cup Ag)^{n-1} \quad (1)$$

where  $n$  is the count of entities in the semantic encyclopedia.

**Table 1.** The fuzzy semantic relations used for the determination of the context

Symbol	Name	Symbol	Name
<i>Sp</i>	Specialization	<i>Pat</i>	Patient
<i>Ct</i>	Context	<i>Loc</i>	Location
<i>Ins</i>	Instrument	<i>Ag</i>	Agent
<i>P</i>	Part		

### 3.2 The Notion of Context

In the processes of video content and user query analysis we utilize the common meaning of semantic entities. Let  $A = \{s_1, s_2, \dots, s_n\}$ , denote a set of semantic entities, and  $S \supseteq A$  be the global set of semantic entities. The common meaning of, and more generally, whatever is common among the elements of  $A$  is their context  $K(A)$ . Assuming that  $A$  is a crisp set, i.e. that no fuzzy degrees of membership are contained, the context of the group, which is again a set of semantic entities, can be defined simply as the set of the common descendants of the members of the set.

$$K(A) = \bigcap_{s_i \in A} T(s_i) \quad (2)$$

In the fuzzy case and assuming that fuzzy set  $A$  is normal, we extend the above definition as follows, where  $c$  is an Archimedean fuzzy complement [10]:

$$K(A) = \bigcap_{s_i \in A} K(s_i) \quad (3)$$

$$K(s_i) = T(s_i) \cup c(A(s_i)) \quad (4)$$

## 4 Analysis and Indexing

### 4.1 Fuzzy Hierarchical Clustering of Semantic Entities

The detection of the topics that are related to a document  $d$  requires that the set of semantic entities that are related to it are clustered, according to their common meaning. Since the number of topics that exist in a document is not known beforehand, partitioning methods are inapplicable for this task [11] and a hierarchical clustering algorithm needs to be applied [12].

The two key points in hierarchical clustering are the identification of the clusters to merge at each step, i.e. the definition of a meaningful metric  $d(c_1, c_2)$  for a pair of clusters  $c_1, c_2$ , and the identification of the optimal terminating step, i.e. the definition of a meaningful termination criterion. From a semantic point of view, two entities are related to the extent that they refer to the same concepts or belong to the same

abstract class. Thus, we utilize as a metric for the comparison of clusters  $c_1$  and  $c_2$  the intensity of their common context:

$$d(c_1, c_2) = h(K(c_1 \cup c_2)) \quad (5)$$

The termination criterion is a threshold on the selected compatibility metric. This clustering method, being a hierarchical one, successfully determines the count of distinct clusters that exist in document  $d$ , but suffers from a major disadvantage; it only creates crisp partitions, not allowing for overlapping of clusters or fuzzy membership degrees. Using the semantic entities contained in each cluster  $c$  we design a fuzzy classifier that is able to generate fuzzy partitions as follows:

$$C_c(s) = \frac{h(K(c \cup s))}{h(K(c))} \quad (6)$$

## 4.2 Thematic Categorization and Detection of Events and Objects

For each cluster of semantic entities detected in document  $d$ , the context of the cluster describes the topics that are related to the entities of the cluster. This information can be utilized both for thematic categorization of the document and for detection of other objects, through simple inference. Thematic categories are semantic entities that have been selected as having a special meaning. The context of the fuzzy clusters is first calculated, using the inverse of relation  $T$  as the base quasi-taxonomy. The thematic categories that are found to belong to the context of the cluster are extracted as the thematic categorization of the cluster. Information obtained from clusters with small cardinality is ignored, as possibly erroneous.

$$TC(d) = \bigcup [TC(c_i) L(|c_i|)] \quad (7)$$

$$TC(c_i) = w(K(c_i) \cap S_{TC}) \quad (8)$$

where  $w$  is a weak modifier and  $S_{TC}$  is the set of thematic categories. On the other hand, the presence of simple events and objects is typically detected to a smaller extent. We infer that a simple semantic entity is present when one of its special cases has been detected:

$$O(d) = \bigcup O(c_i) \quad (9)$$

$$O(c_i) = \{s_j / \min(h(T(s_j) \cap K(c_i)), h(T(s_l), \{s_j\})))\}, s_l \in c_i \quad (10)$$

Both results are used to enrich the initial indexing of the document. Thus, if  $I(d)$  is the fuzzy set of entities to which the document is linked, the set is extended as follows:

$$I'(d) = I(d) \cup TC(d) \cup O(d) \quad (11)$$

## 5 Video Retrieval

### 5.1 Query Analysis

#### 5.1.1 Context-Sensitive Query Interpretation

Ideally, a user query consists of keywords that correspond to one semantic entity. In some cases though, this is not true and some words can be matched to more than one semantic entity. It is left to the system to make the decision, based on the context of the query, which semantic entity was implied by the user. However, the detection of the query context cannot be performed before the query interpretation is completed, which in turn needs the result of the query context mining. Therefore both tasks must be done simultaneously. Let the textual query contain the terms  $t_i$ ,  $i=1,2,\dots$ . Let also  $\tau_i$  be the textual description of semantic entities  $s_{ij}$ ,  $j=1,2,\dots,M_i$ . Then there exist  $N_Q = \prod_i M_i$  distinct combinations of semantic entities that may be used for the

representation of the user query. Out of the candidate queries  $q_k$ ,  $k = 1,2,\dots,N_Q$ , the one that has the most intense context is selected:

$$q = q_i \in \{q_1, \dots, q_{N_Q}\} : h(q_i) \geq h(q_j) \forall q_j \in \{q_1, \dots, q_{N_Q}\} \quad (12)$$

#### 5.1.2 Context-Sensitive Query Expansion

Query expansion enriches the query in order to increase the probability of a match between the query and the document index. The presence of several semantic entities in the query created during the query interpretation defines a context, which is used to direct the expansion process.

More formally, we replace each semantic entity  $s_i$  with a set of semantic entities  $X(s_i)$ ; we will refer to this set as the expanded semantic entity. In a context-sensitive query expansion, the degree of significance,  $x_{ij}$ , of the entity  $s_j$  in the expanded semantic entity  $X(s_i)$  is not only proportional to the weight  $w_i$ , but depends on the degree of the relation  $T(s_i, s_j)$  as well. We define the measure of relevance as:

$$h_j = \max\left(\frac{h(T(s_j) \cap K(q))}{h(K(q))}, c(h(K(q)))\right) \quad (13)$$

The fuzzy complement  $c$  in this relation is Yager's complement with a parameter of 0.5. Considering now the initial entity's importance in the query and the degree to which the initial and the candidate entity are related, we have

$$x_{ij} = h_j q(s_i) T(s_i, s_j) \quad (14)$$

## 5.2 Index Matching

When querying, we treat  $X(s_i)$  considering a union operation, i.e. documents that match any entity contained in  $X(s_i)$  are selected. If  $I$  is the semantic index in the form of a fuzzy relation from semantic entities to documents, then the set of documents  $R_i$  that match extended entity  $X(s_i)$  is calculated as

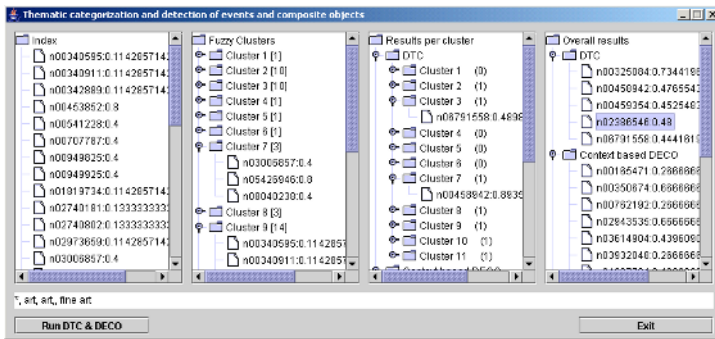
$$R_i = X(s_i) \circ I \quad (15)$$

On the other hand, results from distinct entities are typically treated using an intersection operator, i.e. only documents that match all of the entities of the query are selected. This is quite limiting; it is more intuitive to return the documents that match all of the terms in the query first, followed by those documents that match less of the query terms. We achieve such an intuitive results utilizing an ordered weighted average operator in order to produce the overall result. Thus, the overall result is calculated as

$$R = OWA(R_1, R_2, \dots) \quad (16)$$

## 6 Results

The methodologies presented herein have been developed and applied to a set of 175 documents from the audiovisual archive of the Hellenic Broadcasting Corporation (ERT) i.e. videos of total duration of 20 hours. The a/v documents were already extensively annotated according to the MPEG-7 standard. In this section we provide some indicative results of the application on both the indexing and retrieval processes.



**Fig. 3.** Results of multimedia document analysis, object detection and thematic categorization.

In Fig. 3 we present an implementation of the document analysis methodologies described in section 4. In the first column, the IDs of the objects detected in the video are presented. In the semantic encyclopaedia each one of these IDs is related to a

textual description, a set of keywords, and possibly a set of audiovisual descriptors. For example, ID n02386546 is related to keywords “art” and “fine art”, as can be seen at the lower part of the application. In the second column the entities have been clustered applying the fuzzy hierarchical clustering algorithm. In column 3, thematic categorization information is extracted from each cluster and some simple objects are detected. Finally, in column 4 results are summarized for all documents. Not all thematic categories detected in distinct clusters participate in the overall result; findings that correspond to clusters of small cardinality have been ignored, as they are possibly misleading. On the contrary, all information for detected objects is kept.

SEMANTIC AND METADATA SEARCH - SemanticResponse			
The expanded set of semantic entities has been matched with the following multimedia documents in the fashion semantic index, with degree of relevance:			
Id	Title	Source/Archive	Score
35	Flugzeugkabinestraße	FAA	0.9
1169	Επισκόπηση Αυτοκινήτου 1974	ERT	0.8
52	Die Vietnamkriege	FAA	0.78
1	Sensationell neuer Rettungsstrategie	FAA	0.72
1514	Πασιονόμο	ERT	0.68
AV2-A-004129-0038	Archaeological excavations in Rome	Alinari	0.6
11	Ausbau und Beibehaltung der Strada Graz-Buch	FAA	0.5
FCC-F-021980-0000	Exodus of the Belgian population	Alinari	0.45

Pages:

<< PREVIOUS

Query Interpretation

Query Expansion

SemanticResponse

PresentationResponse

ClassificationResponse

NEXT >>

Fig. 4. Ranked multimedia documents retrieved for a user query with the keyword “politics”.

Fig. 4 demonstrates the results of a user query corresponding to the keyword “politics”. Thematic categorization and detection of events and objects has been performed beforehand. The search process starts with query interpretation and expansion. Index matching is then performed and the list of the retrieved documents is ranked according to their degree of relevance to the semantic entity “politics”. As the data set was used for the adjustment of the knowledge (in example for the thematic categories extraction), the results during the evaluation process were remarkably good. As future work we intend to extend the data set and conduct further evaluations in a more generalized set of documents.

7 Conclusions

In this paper we have discussed the utilization of semantic knowledge for the analysis and retrieval of video. Specifically, we have followed a fuzzy relational approach to knowledge representation, based on which we have defined and extracted the context, i.e. the common overall meaning, of a set of semantic entities. The notion of context has then been applied in the understanding of both the user and the video content.

As far as the analysis of video is concerned, the context of simple objects detected in its content has been used to detect the topics to which the video is related. From those topics we perform thematic categorization and detect simple objects that were not identified in the video but their presence can be inferred. As far as the user is concerned, the context of the user query is used to clarify the exact meaning of the query terms and to meaningfully guide the process of query expansion. Results have been provided that are indicative of the proposed approach.

## References

1. Koenen R.: "Overview of the MPEG-4 Standard", ISO/IEC JTC 1/SC 29/WG 11/N4668, March 2002
2. T. Sikora. The MPEG-7 Visual standard for content description - an overview. *IEEE Trans. on Circuits and Systems for Video Technology*, 11(6):696–702, June 2001
3. Y. Avrithis, G. Stamou, A. Delopoulos and S. Kollias: Intelligent Semantic Access to Audiovisual Content. In *Proc. of 2nd Hellenic Conference on Artificial Intelligence (SETN'02)*, Thessaloniki, Greece, April 11-12, 2002
4. J. Hunter. Adding Multimedia to the Semantic Web: Building an MPEG-7 Ontology. In *Proc. The First Semantic Web Working Symposium (SWWS'01)*, Stanford University, California, USA, July 2001
5. A. Smeulders, M. Worring, and S. Santini: Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(12), 2000
6. M. La Cascia, S. Sethi, and S. Sclaroff: Combining textual and visual cues for content-based image retrieval on the world wide web. *IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL'98)*, June 1998
7. G. Stamou, Y. Avrithis, S. Kollias, F. Marques and P. Salembier: Semantic Unification of Heterogenous Multimedia Archives. In *Proc. of 4th European Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'03)*, London, UK, April 9-11, 2003
8. P. Tzouveli, G. Andreou, G. Tsechpenakis, Y. Avrithis and S. Kollias, "Intelligent Visual Descriptor Extraction from Video Sequences," in *Proc. of 1st International Workshop on Adaptive Multimedia Retrieval (AMR '03)*, Hamburg, Germany, September 15-18, 2003
9. Wallace, M., Akrivas, G. and Stamou, G.: Automatic Thematic Categorization of Documents Using a Fuzzy Taxonomy and Fuzzy Hierarchical Clustering. In *Proc. of IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, St. Louis, MO, USA, May 2003
10. Klir G. and Bo Yuan, *Fuzzy Sets and Fuzzy Logic, Theory and Applications*, New Jersey, Prentice Hall, 1995
11. Miyamoto S.: *Fuzzy sets in information retrieval and cluster analysis*. Kluwer Academic publishers, 1990
12. S. Theodoridis and K. Koutroubas: *Pattern Recognition*. Academic Press, 1998

# Using Structure for Video Object Retrieval

Lukas Hohl, Fabrice Souvannavong, Bernard Merialdo, and Benoit Huet

Multimedia Department  
Institute Eurecom  
2229 routes des Cretes  
06904 Sophia-Antipolis, France  
{hohl, souvanna, merialdo, huet}@eurecom.fr

**Abstract.** The work presented in this paper aims at reducing the semantic gap between low level video features and semantic video objects. The proposed method for finding associations between segmented frame region characteristics relies on the strength of Latent Semantic Analysis (LSA). Our previous experiments [1], using color histograms and Gabor features, have rapidly shown the potential of this approach but also uncovered some of its limitation. The use of structural information is necessary, yet rarely employed for such a task. In this paper we address two important issues. The first is to verify that using structural information does indeed improve performance, while the second concerns the manner in which this additional information is integrated within the framework. Here, we propose two methods using the structural information. The first adds structural constraints indirectly to the LSA during the preprocessing of the video, while the other includes the structure directly within the LSA. Moreover, we will demonstrate that when the structure is added directly to the LSA the performance gain of combining visual (low level) and structural information is convincing.

## 1 Introduction

Multimedia digital documents are readily available, either through the internet, private archives or digital video broadcast. Traditional text based methodologies for annotation and retrieval have shown their limit and need to be enhanced with content based analysis tools. Research aimed at providing such tools have been very active over recent years [2]. Whereas most of these approaches focus on frame or shot retrieval, we propose a framework for effective retrieval of semantic video objects. By video object we mean a semantically meaningful spatio-temporal entity in a video.

Most traditional retrieval methods fail to overcome two well known problems called synonymy and polysemy, as they exist in natural language. Synonymy causes different words describing the same object, whereas polysemy allows a word to refer to more than one object. Latent Semantic Analysis (LSA) provides a way to weaken those two problems [3]. LSA has been primarily used in the field of natural language understanding, but has recently been applied to domains such as source code analysis or computer vision. Latent Semantic Analysis has also provided very promising results in finding the semantic meaning of multimedia documents [1,4,5]. LSA is based on a Singular Value Decomposition (SVD) on a word by context matrix, containing the frequencies of occurrence of words in each context. One of the limitations of the LSA is that it does



not take into account word order, which means it completely lacks the syntax of words. The analysis of text, using syntactical structure combined with LSA already has been studied [6,7] and has shown improved results. For our object retrieval task, the LSA is computed over a visual dictionary where region characteristics, either structurally enhanced or not, correspond to words.

The most common representation of visual content in retrieval system relies on global low level features such as color histograms, texture descriptors or feature points, to name only a few [8,9,10,11]. These techniques in their basic form are not suited for object representation as they capture information from the entire image, merging characteristics of both the object and its surrounding, in other word the object description and its surrounding environment become merged. A solution is to segment the image in regions with homogenous properties and use a set of low level features of each region as global representation. In such a situation, an object is then referred to as a set of regions within the entire set composing the image. Despite the obvious improvement over the global approach, region based methods still lack important characteristics in order to uniquely define objects. Indeed it is possible to find sets of regions with similar low level features yet depicting very different content. The use of relational constraints, imposed by the region adjacency of the image itself, provides a richer and more discriminative representation of video object. There has only been limited publications employing attributed relational graph to describe and index into large collection of visual data [12, 13,14,15] due to the increased computational complexity introduced by such approaches. Here we will show that it is possible to achieve significant performance improvement using structural constraints without increasing either the representation dimensionality or the computational complexity.

This paper is organized as follows. The concept of adding structure to LSA and a short theoretical background on the algorithms used, are presented in Section 2. Section 3 provides the experimental results looking at several different aspects. The conclusion and future directions are discussed in Section 4.

## **2 Enhancing Latent Semantic Analysis with Structural Information**

As opposed to text documents there is no predefined dictionary for multimedia data. It is therefore necessary to create one to analyze the content of multimedia documents using the concept of Latent Semantic Analysis [3]. Here, we propose three distinct approaches for the construction of visual dictionaries. In the non-structural approach, each frame region of the video is assigned to a class based on its properties. This class corresponds to a "visual" word and the set of all classes is our visual dictionary. In the case where we indirectly add structure, the clustering process which builds the different classes (words) takes structural constraints into account. Finally, in the third case where structure is added directly to the LSA, pairs of adjacent regions classes (as in the non-structural approach) are used to define words of the structural dictionary. We shall now detail the steps leading to three different dictionary constructions.

## 2.1 Video Preprocessing

We consider a video  $V$  as a finite set of frames  $\{F_1, \dots, F_n\}$ , where the preprocessing is performed on subsampled individual frames. Such an approach implies that video scenes and/or shots are not taken into account. Every 25th frame of the video  $V$  is segmented in regions  $R_i$  using the method proposed by Felzenszwalb and Huttenlocher in [16]. This algorithm was selected for its perceived computation requirement and segmentation quality ratio. Each segmented region  $R_i$  is characterized by its attributes, feature vectors that contain visual information about the region such as color, texture, size or spatial information. For this paper, the feature vector is limited to a 32 bin color histogram of the corresponding region. Other attributes could indeed lead to better results, however for the scope of this paper we are only interested in identifying whether structural constraint provide performance improvements.

## 2.2 Building the Basic Visual Dictionary

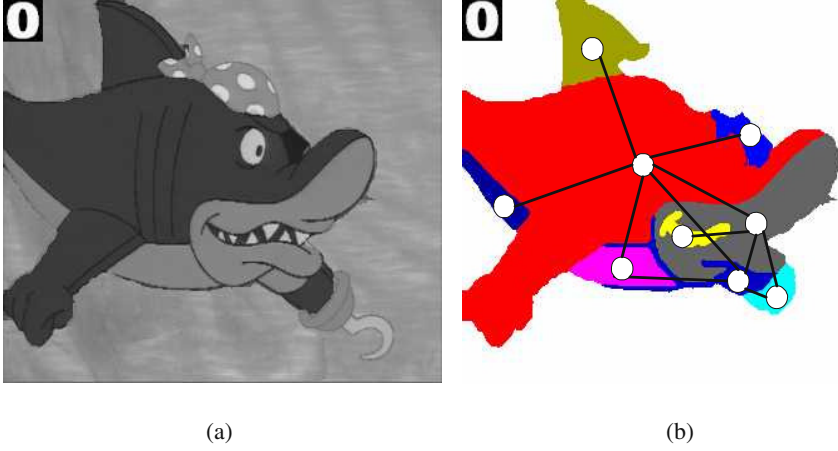
The structure-less dictionary is constructed by grouping regions with similar feature vectors together. There are many ways to do so [17]. Here the k-means clustering algorithm [17] is employed with the Euclidean distance as similarity measure. As a result each region  $R_i$  is mapped to a cluster  $C_l$  (or class), represented by its cluster centroid. Thanks to the k-means clustering parameter  $k$  controlling the number of clusters, the dictionary size may be adjusted to our needs. In this case, each cluster represents a word for the LSA.

## 2.3 Incorporating Structural Information

In an attempt to increase the influence of local visual information, an adjacency graph is constructed from the segmented regions for each frame. Nodes in the graph represent segmented regions and are attributed with a vector  $H$ . Vertices between two nodes of the graph correspond to adjacent regions. A segmented frame can therefore be represented as a graph  $G = (V, E)$  consisting of a set of vertices  $V = \{v_1, v_2, \dots, v_n\}$  and edges  $E = \{e_1, e_2, \dots, e_m\}$ , where the vertices represent the cluster number labelled regions and the edges the connectivity of the regions. For the discussion below, we also introduce  $\phi_i^Q = \{h | (i, h) \in E^Q\}$  which denotes all the nodes connected to a given node  $i$  in a graph  $Q$ . As an illustration, Figure 1(b) shows a frame containing an object segmented into regions with its corresponding relational graph overlaid.

### Indirectly Adding Structure When Building the Dictionary

A first approach to add structural information when using LSA is to include the structural constraints within the clustering process itself. Here we are interested in clustering regions according to their attributes as well as the attributes of the regions they are adjacent to. To this end, we used a clustering algorithm similar to k-medoid with a specific distance function  $D(R_i^Q, R_j^D)$  (1). This distance function between regions  $R_i^Q$  of graph  $Q$  and  $R_j^D$  of graph  $D$  take the local structure into account.



**Fig. 1.** (a) The shark object and (b) its corresponding graph of adjacent regions.

$$D(R_i^Q, R_j^D) = L_2(H_i, H_j) + \frac{1}{\|\phi_i^Q\|} \sum_{k \in \phi_i^Q} \min_{l \in \phi_j^D} L_2(H_k, H_l) \quad (1)$$

where  $L_2(H_i, H_j)$  is the Euclidian distance between histograms  $H_i$  and  $H_j$ . In order to deal with the different connectivity levels of nodes, the node with the least number of neighbours is  $\phi_l^Q$ . This insures that all neighbour from  $\phi_l^Q$  can be mapped to nodes of  $\phi_j^D$ . Note that this also allows multiple mappings, which means that several neighbours of one node  $i$  can be mapped to the same neighbour of the node  $l$ .

As a result of the clustering described above, we get  $k$  clusters, which are built upon structural constraints and visual features. Each region  $R_i$  belongs to one cluster  $C_l$ . Each cluster represents a visual word for the Latent Semantic Analysis.

### Adding Structural Constraints Directly to the Words of the Dictionary

We now wish to construct a visual dictionary  $D_\nu$  (of size  $\nu$ ) which is containing words with direct structural information. This is achieved by considering every possible un-ordered pair of clusters as a visual word  $W$ , e.g.  $C_3C_7 \equiv C_7C_3$ . Note that for example the cluster pair  $C_1C_1$  is also a word of the dictionary, since two adjacent regions can fall into the same cluster  $C_l$  despite having segmented them into different regions before.

$$D_\nu = \{W_1, \dots, W_\nu\}$$

$$(C_1C_1) \simeq W_1, (C_1C_2) \simeq W_2, \dots, (C_kC_k) \simeq W_\nu$$

The size  $\nu$  of the dictionary  $D_\nu$  is also controlled by the clustering parameter  $k$  but this time indirectly.

$$\nu = \frac{k \cdot (k-1)}{2} + k \quad (2)$$

To be able to build these pairs of clusters (words), each region is labelled with the cluster number it belongs to (e.g.  $C_{14}$ ). If two regions are adjacent, they are linked in an abstract point of view, which results in a graph  $G_i$  as described previously. Every Graph  $G_i$  is described by its adjacency matrix. The matrix is a square matrix ( $n \times n$ ) with both, rows and columns, representing the vertices from  $v_1$  to  $v_n$  in an ascending order. The cell  $(i, j)$  contains the number of how many times vertex  $v_i$  is connected to vertex  $v_j$ . The matrices are symmetric to theirs diagonals.

In this configuration, the LSA is also used to identify which structural information should be favoured in order to obtain good generalisation results. Moreover, we believe that this should improve the robustness of the method to segmentation differences among multiple views of the same object (leading to slightly different graphs).

## 2.4 Latent Semantic Analysis

The LSA describes the semantic content of a context by mapping words (within this context) onto a semantic space. Singular Value Decomposition (SVD) is used to create such a semantic space. A co-occurrence matrix  $\mathbf{A}$  containing words (rows) and contexts (columns) is built. The value of a cell  $a_{ij}$  of  $\mathbf{A}$  contains the number of occurrence of the word  $i$  in the context  $j$ . Then, SVD is used to decompose the matrix  $\mathbf{A}$  (of size  $M \times N$ ,  $M$  words and  $N$  contexts) into three separate matrices.

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T \quad (3)$$

The matrix  $\mathbf{U}$  is of size  $M \times L$ , the matrix  $\mathbf{S}$  is of dimension  $L \times L$  and the matrix  $\mathbf{V}$  is  $N \times L$ .  $\mathbf{U}$  and  $\mathbf{V}$  are unitary matrices, thus  $\mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{I}_L$  where  $\mathbf{S}$  is a diagonal matrix of size  $L = \min(M, N)$  with singular values  $\sigma_1$  to  $\sigma_L$ , where

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_L \quad \mathbf{S} \approx \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_L)$$

$\mathbf{A}$  can be approximated by reducing the size of  $\mathbf{S}$  to some dimensionality of  $k \times k$ , where  $\sigma_1, \sigma_2, \dots, \sigma_k$  are the  $k$  highest singular values.

$$\hat{\mathbf{A}} = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^T \quad (4)$$

By doing a reduction in dimensionality from  $L$  to  $k$ , the sizes of the matrices  $\mathbf{U}$  and  $\mathbf{V}$  have to be changed to  $M \times k$  respectively  $N \times k$ . Thus,  $k$  is the dimension of the resulting semantic space. To measure the result of the query, the cosine measure ( $m_c$ ) is used. The query vector  $\mathbf{q}$  contains the words describing the object, in a particular frame where it appears.

$$\mathbf{q}^T \hat{\mathbf{A}} = \mathbf{q}^T \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^T = (\mathbf{q}^T \mathbf{U}_k) (\mathbf{S}_k \mathbf{V}_k^T) \quad (5)$$

Let  $\mathbf{p}_q = \mathbf{q}^T \mathbf{U}_k$  and  $\mathbf{p}_j$  to be the  $j$ -th context (frame) of  $(\mathbf{S}_k \mathbf{V}_k^T)$

$$m_c(\mathbf{p}_j, \mathbf{q}) = \frac{\mathbf{p}_q \cdot \mathbf{p}_j}{\|\mathbf{p}_q\| \cdot \|\mathbf{p}_j\|} \quad (6)$$

The dictionary size ought to remain "small" to compute the SVD as its complexity is  $O(P^2 k^3)$ , where  $P$  is the number of words plus contexts ( $P = N + M$ ) and  $k$  the number of LSA factors.

### 3 Experimental Results

Here, our object retrieval system is evaluated on a short cartoon (10 minutes duration) taken from the MPEG7 dataset and created by D'Ocon Film Productions. A ground truth has been created by manually annotating frames containing some objects (shown in Figure 2) through the entire video. The query objects are chosen as diverse as possible and appear in 30 to 108 frames of the subsampled video. The chosen granularity of the segmentation results in an average of about 35 regions per frame. Thus the built graphs remain reasonable small, whereas the number of graphs (one per frame) is quite large.

A query object may be created by selecting a set of region from a video frame. Once the query is formed, the algorithm starts searching for frames which contain the query object. The query results are ordered so that the frame which most likely contains the query object (regarding the cosine measure  $m_c$ ) comes first. The performance of our retrieval system is evaluated using either the standard precision vs. recall values or the mean average precision value. The mean average precision value for each object is defined as followed: We take the average precision value obtained after each relevant frame has been retrieved and take the mean value, over all frames retrieved. We have selected 4 objects (Figure 2) from the sequence. Some are rather simple with respect to the number of regions they consist of, while others are more complex. Unless stated otherwise, the plots show the average (over 2 or 4 objects) precision values at given standard recall values [0.1, 0.2, ..., 1.0].

#### 3.1 Impact of the Number of Clusters

To show the impact on the number of clusters chosen during video preprocessing, we have built several dictionaries containing non-structural visual words (as described in Section 2.2). Figure 3(a) shows the precision/recall curves for three cluster sizes (32, 528, 1000). The two upper curves (528 and 1000 clusters) show rather steady high precision values for recall value smaller than 0.6. For 32 clusters the performance results are weaker. Using 528 clusters always delivers as good results as using 1000 clusters which indicates that after a certain number of clusters, the performance cannot be improved and may even start to decay. This is due to the fact that for large  $k$  the number of regions per cluster become smaller, meaning that similar content may be assigned to different clusters.

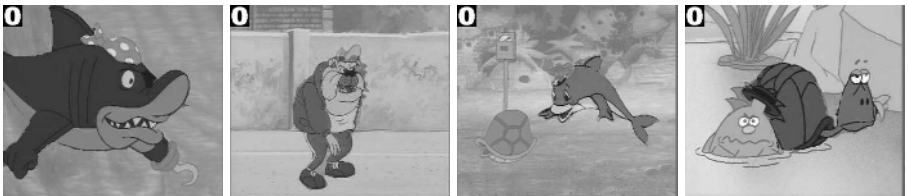
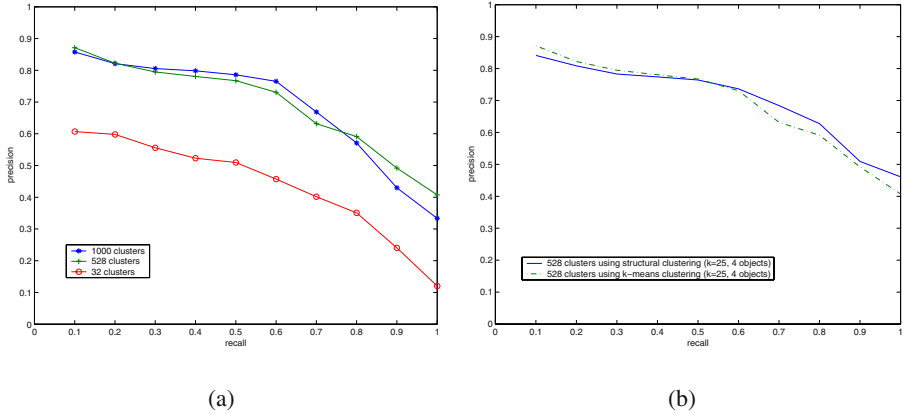


Fig. 2. The 4 query objects.



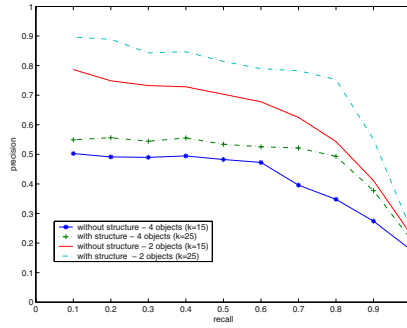
**Fig. 3.** (a) Retrieval performance w.r.t. number of clusters. (b) Retrieval performance for 4 objects queries with indirectly added structure and without.

### 3.2 Comparing Indirectly Added Structure with the Non-structural Approach

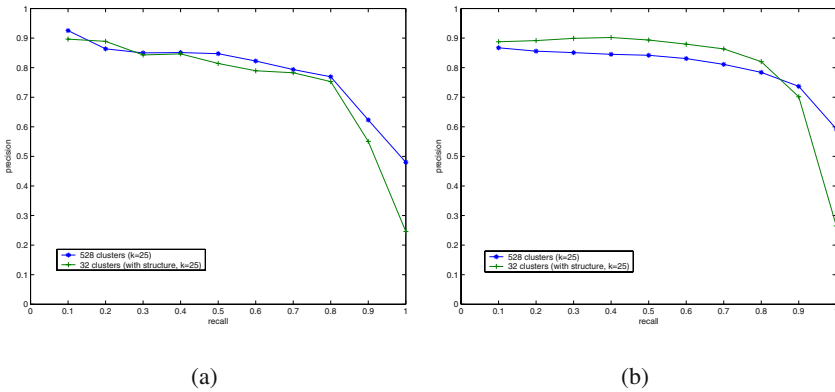
In the following experiment, we compared the retrieval results either using a structure-less dictionary and a dictionary where we added the structural information within the clustering process as explained in Section 2.3. In both methods we use a cluster size of 528 (which also results in a dictionary size of 528) and we select the  $k$  (factor kept in LSA) so that we get best results (in this case  $k=25$ ). Figure 3(b) shows the precision at given recall values for both cases. The curves represent an average over all 4 objects. It shows that adding structural information to the clustering does not improve the non-structural approach, it even is doing slightly worse for recall values above 0.5.

### 3.3 Comparing Directly Structure Enhanced Words with Non-structural Words

For a given cluster size ( $k=32$ ) we compared two different ways of defining the visual words used for LSA. In the non-structural case, each cluster label represents one word, leading to a dictionary size of 32 words. In the structural case, every possible pair of cluster label is defining a word (as explained in Section 2.3), so that the number of words in the dictionary is 528. Note that by building those pairs of cluster labelled regions, there might be some words which never occur throughout all frames of the video. In the case of a cluster size of 32, there will be 14 lines in the co-occurrence matrix which are all filled with zeros. Figure 4 shows the results for both approaches when querying for four objects and two objects. The group of two objects contains the most complex ones. The structural approach clearly outperforms the non-structural methods. Even more so, as the objects are most complex. The structural approach is constantly delivering higher precision values than the non-structural version, throughout the whole recall range.



**Fig. 4.** Retrieval performance for 2 and 4 objects queries with directly added structure and without.



**Fig. 5.** Comparing the structural versus non-structural approach in respect of the same dictionary size looking at 2 objects(a) and looking at the shark(b).

### 3.4 Structure Versus Non-structure for the Same Size of the Dictionary

Here we are looking at both, the direct structural (as explained in Section 2.3) and the non-structural approach, in respect of a unique dictionary size. To this aim, we choose 528 clusters (which equals 528 words) for the non-structural method and 32 clusters for the structural, which results in 528 words as well. In this case we feed the same amount of information to the system for both cases, however the information is of different kind. Figure 5(a) shows the precision/recall values when we look at 2 different objects. The results show that there is no significant improvement of one approach over the other. Overall the non-structural approach is only doing slightly better. However, when looking at one particular object (the shark in this case, see Figure 5(b)), the structural approach is doing constantly better (except for very high recall values 0.9 to 1.0). As mentioned previously, the shark is a highly complex object and therefore it is not surprising that the structural method delivers better results than the non-structural one.

## 4 Conclusion and Future Work

In this paper we have presented two methods for enhancing a LSA based video object retrieval system with structural constraints (either direct or indirect) obtained from the object visual properties. The methods were compared to a similar method [1] which did not make use of the relational information between adjacent regions. Our results show the importance of structural constraints for region based object representation. This is demonstrated in the case where the structure is added directly in building the words, by a 18% performance increase in the optimal situation for a common number of region categories. We are currently investigating the sensitivity of this representation to the segmentation process as well as other potential graph structures.

## References

1. Souvannavong, F., Merialdo, B., Huet, B.: Video content modeling with latent semantic analysis. In: Third International Workshop on Content-Based Multimedia Indexing. (2003)
2. TREC Video Retrieval Workshop (TRECVID) <http://www-nlpir.nist.gov/projects/trecvid/>
3. Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., Harshman, R.A.: Indexing by latent semantic analysis. *American Soc. of Information Science Journal* **41** (1990) 391–407
4. Zhao, R., Grosky, W.I.: Video Shot Detection Using Color Anglogram and Latent Semantic Indexing: From Contents to Semantics. CRC Press (2003)
5. Monay, F., Gatica-Perez, D.: On image auto-annotation with latent space models. *ACM Int. Conf. on Multimedia* (2003)
6. Wiemer-Hastings, P.: Adding syntactic information to lsa. In: Proceedings of the Twenty-second Annual Conference of the Cognitive Science Society. (2000) 989–993
7. Landauer, T., Laham, D., Rehder, B., Schreiner, M.: How well can passage meaning be derived without using word order. *Cognitive Science Society*. (1997) 412–417
8. Swain, M., Ballard, D.: Indexing via colour histograms. *ICCV* (1990) 390–393
9. M. Flickner, H. Sawhney, e.a.: Query by image and video content: the qbic system. *IEEE Computer* **28** (1995) 23–32
10. Pentland, A., Picard, R., Sclaroff, S.: Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision* **18** (1996) 233–254
11. Gimelfarb, G., Jain, A.: On retrieving textured images from an image database. *Pattern Recognition* **29** (1996) 1461–1483
12. Shearer, K., Venkatesh, S., Bunke, H.: An efficient least common subgraph algorithm for video indexing. *International Conference on Pattern Recognition* **2** (1998) 1241–1243
13. Huet, B., Hancock, E.: Line pattern retrieval using relational histograms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **21** (1999) 1363–1370
14. Sengupta, K., Boyer, K.: Organizing large structural modelbases. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. (1995)
15. Messmer, B., Bunke, H.: A new algorithm for error-tolerant subgraph isomorphism detection. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. (1998)
16. Felzenszwalb, P., Huttenlocher, D.: Efficiently computing a good segmentation. In: *IEEE Conference on Computer Vision and Pattern Recognition*. (1998) 98–104
17. Jain, A.K., Dubes, R.C.: *Algorithms for Clustering Data*. Prentice Hall, Englewood Cliffs, NJ (1988)



# Object Segmentation and Ontologies for MPEG-2 Video Indexing and Retrieval\*

Vasileios Mezaris<sup>1,2</sup> and Michael G. Strintzis<sup>1,2</sup>

<sup>1</sup> Information Processing Laboratory, Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece

<sup>2</sup> Informatics and Telematics Institute (ITI)/ Centre for Research and Technology Hellas (CERTH), Thessaloniki 57001, Greece

**Abstract.** A novel approach to object-based video indexing and retrieval is presented, employing an object segmentation algorithm for the real-time, unsupervised segmentation of compressed image sequences and simple ontologies for retrieval. The segmentation algorithm uses motion information directly extracted from the MPEG-2 compressed stream to create meaningful foreground spatiotemporal objects, while background segmentation is additionally performed using color information. For the resulting objects, MPEG-7 compliant low-level indexing descriptors are extracted and are automatically mapped to appropriate intermediate-level descriptors forming a simple vocabulary termed *object ontology*. This, combined with a relevance feedback mechanism, allows the qualitative definition of the high-level concepts the user queries for (*semantic objects*, each represented by a *keyword*) and the retrieval of relevant video segments. Experimental results demonstrate the effectiveness of the proposed approach.

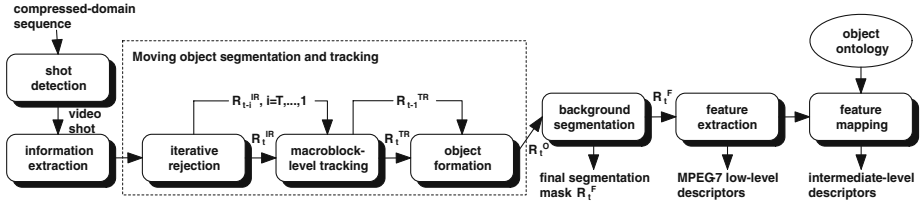
## 1 Introduction

Sophisticated query and retrieval from video databases is an important part of many emerging multimedia applications. Retrieval schemes of such applications employ descriptors ranging from low-level features to higher-level semantic concepts. In all cases, preprocessing of video data is necessary as the basis on which indices are extracted. The preprocessing is of *coarse granularity* if it involves processing of video frames as a whole, whereas it is of *fine granularity* if it involves detection of objects within a video frame [1]. In this work, a fine granularity approach is adopted.

To this end, several approaches have been proposed in the literature for video segmentation. Most of these operate in the uncompressed pixel domain [2], which enables object boundary estimation with pixel accuracy but requires that the sequence be fully decoded before segmentation. As a result, the usefulness of such

---

\* This work was supported by the EU project SCHEMA “Network of Excellence in Content-Based Semantic Scene Analysis and Information Retrieval” (IST-2001-32795).



**Fig. 1.** Overview of the compressed-domain spatiotemporal segmentation algorithm and the feature extraction procedure.

approaches is usually restricted to non-real-time applications. To counter these drawbacks, compressed domain methods have been proposed for spatiotemporal segmentation and indexing [3,4]. Although significantly faster than most pixel-domain algorithms, some of them cannot operate in real-time [5].

To allow efficient indexing of large video databases, an algorithm for the real-time, unsupervised spatiotemporal segmentation of MPEG-2 video sequences is proposed. Only I- and P-frames are examined, since they contain all information that is necessary for the proposed algorithm; this is also the case for most other compressed-domain algorithms. Both foreground and background spatiotemporal objects are identified for each shot.

In the proposed indexing and retrieval scheme, instead of adopting the query-by-example strategy, the aforementioned segmentation algorithm is combined with simple *ontologies* [6] and a *relevance feedback* mechanism. This scheme (Fig. 1) allows for MPEG-7 compliant low-level indexing features to be extracted for the spatiotemporal objects and subsequently be associated with higher-level descriptors that humans are more familiar with; these are used to restrict the search to a set of potentially relevant spatiotemporal objects. Final query results are produced after one or more rounds of relevance feedback, similarly to [6] where this method was applied to still images.

The paper is organized as follows: in section 2 video object segmentation and tracking methods are developed. The indexing and retrieval scheme is discussed in section 3. Section 4 contains experimental results and finally, conclusions are drawn in section 5.

## 2 Video Object Segmentation and Tracking

### 2.1 Compressed-Domain Information Extraction

The information used by the proposed segmentation algorithm is extracted from MPEG-2 sequences during the decoding process. Specifically, motion vectors are extracted from the P-frames and are used for foreground/background segmentation and for the subsequent identification of different foreground objects. In order to derive motion information for the I-frames, averaging of the motion

vectors of the P-frames that are temporally adjacent to the given I-frame is performed. Additionally, color information is used in order to further segment the background to its constituent objects. The employed color information is restricted to the DC coefficients of the I-frame macroblocks, corresponding to the Y, Cb and Cr components of the MPEG color space.

## 2.2 Iterative Macroblock Rejection

Iterative rejection for the estimation of the eight parameters of the bilinear motion model was proposed in [7], where it was used for the retrieval of video clips based on their global motion characteristics. This method is based on iteratively estimating the parameters of the global-motion model using least-square estimation and rejecting those blocks whose motion vectors result in larger than average estimation errors, the underlying assumption being that the background is significantly larger than the area covered by the moving objects.

In this work, iterative rejection based on the bilinear motion model is used to generate the mask  $R_t^{IR}$ , indicating which macroblocks have been rejected at time  $t$  (or activated, from the segmentation objective's point of view). This is the first step of foreground / background segmentation. Rejected (activated) macroblocks are treated as potentially belonging to foreground objects.

## 2.3 Macroblock-Level Tracking

In order to examine the temporal consistency of the output of iterative rejection, activated macroblocks are temporally tracked using the compressed-domain motion vectors. The temporal tracking is based upon the work presented in [8], where objects are manually marked by selecting their constituent macroblocks and are subsequently tracked. However, in contrast to the method in [8], the proposed method requires no human intervention for the selection of the macroblocks to be tracked. A shortcoming of the former method, the need for block matching in order to extract motion features for the I-frames, is avoided in the present work by averaging the motion vectors of the P-frames that are temporally adjacent to the given I-frame.

More specifically, let  $\tau(\cdot)$  be the tracking operator realizing the tracking process of [8], whose input is a macroblock at time  $t$  and its output is the corresponding macroblock or macroblocks at time  $t + 1$ . This correspondence is established by estimating the overlapping of the examined macroblock with its spatially adjacent ones, determined using the displacement indicated by its motion vector. Then, the operator  $\mathcal{T}(\cdot)$  is defined as having a mask (such as  $R_t^{IR}$ ) as input, applying the  $\tau(\cdot)$  operator to the set of all foreground macroblocks of that mask, and outputting the corresponding mask at time  $t + 1$ .

Let  $R_t^{TR}$  denote the output foreground/background mask derived via macroblock level tracking, using masks  $R_{t-i}^{IR}$ ,  $i = T, \dots, 0$ . The derivation of mask  $R_t^{TR}$ , using the operator  $\mathcal{T}(\cdot)$  to evaluate and enforce the temporal consistency of the output of iterative rejection over  $T$  frames, can be expressed as:

$$R_{t-T}^{temp} = R_{t-T}^{IR}$$

$$\text{for } i = T, \dots, 1, \quad R_{t-i+1}^{temp} = \mathcal{T}(R_{t-i}^{temp}) \cap R_{t-i+1}^{IR} \\ R_t^{TR} = R_t^{temp}$$

where  $\cap$  denotes the intersection of foreground macroblocks and  $R_{t-i}^{temp}$ ,  $i = T, \dots, 0$  is a set of temporary foreground/background segmentation masks.

## 2.4 Spatiotemporal Object Formation

After the rejection of falsely activated macroblocks, as described in the previous subsection, the remaining macroblocks are clustered to connected foreground regions and are subsequently assigned to foreground spatiotemporal objects. Clustering to connected regions  $s_k^t$ ,  $k = 1, \dots, \kappa^t$  using a four-connectivity component labelling algorithm results in the creation of mask  $R_t^I$ .

To determine whether a given spatial region of  $R_t^I$  belongs to one or more pre-existing spatiotemporal objects or to a newly appearing one, and to eventually create the object mask  $R_t^O$ , motion projection is performed by applying the tracking operator  $\tau(\cdot)$  to the macroblocks of each spatiotemporal object of mask  $R_{t-1}^O$ . Thus, every connected region  $s_k^t$  of mask  $R_t^I$  can be assigned to one of the following three categories:

- Cat. 1. A number of macroblocks  $M_{k,q}$ ,  $M_{k,q} \geq S_{k,q}$ , of  $s_k^t$  have been assigned to spatiotemporal object  $o_q$  in mask  $\mathcal{T}(R_{t-1}^O)$ , and no macroblock of  $s_k^t$  has been assigned to a spatiotemporal object  $o_m$ ,  $m \neq q$ .
- Cat. 2. A number of macroblocks  $M_{k,q}$ ,  $M_{k,q} \geq S_{k,q}$ , of  $s_k^t$  have been assigned to spatiotemporal object  $o_q$  in mask  $\mathcal{T}(R_{t-1}^O)$ , and one or more macroblocks of  $s_k^t$  have been assigned to different spatiotemporal objects, namely  $o_m$ ,  $m = 1, \dots, M$ .
- Cat. 3. There is no spatiotemporal object  $o_q$  in mask  $\mathcal{T}(R_{t-1}^O)$  having  $M_{k,q}$  macroblocks of  $s_k^t$ ,  $M_{k,q} \geq S_{k,q}$ , assigned to it.

The parameter  $S_{k,q}$  in the definition of the above categories is estimated for every pair of a spatial region  $s_k^t$  of mask  $R_t^I$  and the projection of a spatiotemporal object  $o_q$  in mask  $\mathcal{T}(R_{t-1}^O)$ . Let  $M_k^s$ ,  $M_q^o$  denote their sizes in macroblocks, then parameter  $S_{k,q}$  is calculated as  $S_{k,q} = a \cdot \frac{M_k^s + M_q^o}{2}$ .

The spatial regions  $s_k^t$  classified in the third category can not be associated with an existing spatiotemporal object; therefore, each region of this category forms a new spatiotemporal object. Similarly, the regions classified in the first category can only be associated with a single spatiotemporal object  $o_q$ ; however, more than one spatial regions may be associated with the same spatiotemporal object. In this case, the larger spatial region becomes part of  $o_q$ , while the rest are discarded. As for the regions  $s_k^t$  classified in the second category, the initial correspondence of specific macroblocks belonging to  $s_k^t$  with spatiotemporal objects is employed so as to estimate the parameters of the bilinear motion model for each of the competing objects. Subsequently, each macroblock is assigned to the object for which the motion estimation error is minimized and possible merging of adjacent objects is examined using the parameters of their common motion model.

## 2.5 Background Segmentation

After foreground spatiotemporal objects have been extracted, background segmentation is performed based on classifying the remaining macroblocks to one of a number of background spatiotemporal objects. Background segmentation begins by applying the *maximin* algorithm to the color DC coefficients of the first I-frame. Then, in I-frames, background macroblocks are clustered to background objects using the K-means algorithm, where  $K$  is the number of objects estimated by the maximin algorithm. In P-frames, the absence of color information is dealt with by using the macroblock motion vectors and a previous final mask  $R_{t-1}^F$ . Temporal tracking is then performed as discussed in previous sections.

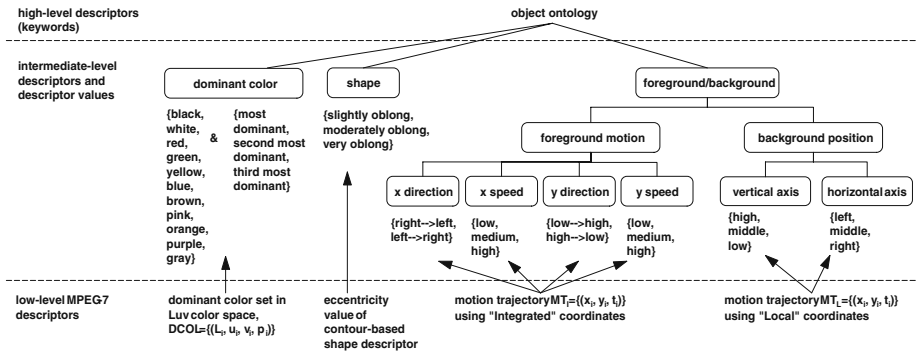
## 3 Object-Based Indexing and Retrieval

### 3.1 Overview

The proposed segmentation algorithm is suitable for introducing object-based functionalities to video indexing and retrieval applications, due to the formation of both foreground and background spatiotemporal objects, for which object-based descriptors in the context of the MPEG-7 Visual standard [9] can be extracted.

With the exception of a few MPEG-7 descriptors, most standardized descriptors are low-level arithmetic ones. When examining the specific application of object-based video indexing, however, it is possible to translate certain low-level arithmetic values to intermediate-level descriptors qualitatively describing the object attributes; the latter are more suitable for manipulation by humans. Extending the approach in [6], these intermediate-level descriptors form a simple vocabulary, the *object ontology*. Ontologies are tools for structuring knowledge, defined as the specification of a representational vocabulary for a shared domain of discourse which may include definitions of classes, relations, functions and other objects. In the proposed scheme, ontologies are used to facilitate the mapping of low-level descriptor values to higher-level semantics.

Under the proposed scheme, a query is initiated by the user qualitatively describing the semantic objects and their relations in the desired shot. By comparing the user-supplied qualitative description with the one automatically estimated for each spatiotemporal object, clearly irrelevant ones can be discarded; the remaining, potentially relevant ones are presented to the user at random order. The user then evaluates a subset of them, marking relevant ones simply by checking the appropriate “relevant” box. By submitting this relevance feedback, a support vector machine is trained and subsequently ranks according to relevance all potentially relevant spatiotemporal objects, using their low-level descriptor values; the shots containing these objects are then presented to the user, ordered by rank. This relevance feedback process can then be repeated, to further enhance the output of the query.



**Fig. 2.** Object ontology. The correspondence between low-level MPEG-7 descriptors and intermediate-level descriptors is shown.

### 3.2 MPEG-7 Descriptors

As soon as a sequence of segmentation masks is produced for each video shot, a set of descriptor values useful in querying the database are calculated for each spatiotemporal object. Standardized MPEG-7 descriptors are used, to allow for flexibility in exchanging indexing information with other MPEG-7 compliant applications. The different MPEG-7 descriptors used in this work are: Motion Activity, Dominant Color, GoF/GoP Color, Contour Shape, Motion Trajectory using “Local” coordinates, and Motion Trajectory using “Integrated” coordinates.

### 3.3 Object Ontology

In this work, ontologies [6] are employed to allow the user to query a video collection using semantically meaningful concepts (semantic objects), without the need for performing manual annotation of visual information. A simple *object ontology* is used to enable the user to describe semantic objects, like “tiger”, using a vocabulary of intermediate-level descriptor values. These are automatically mapped to the low-level descriptor values calculated for each spatiotemporal object in the database, thus allowing the association of keywords representing semantic objects (e.g. the “tiger” keyword) and potentially relevant spatiotemporal objects. The simplicity of the employed object ontology permits its applicability to generic video collections without requiring the correspondence between spatiotemporal objects and relevant descriptors to be defined manually. This object ontology can be expanded so as to include additional descriptors corresponding either to low-level properties (e.g. texture) or to higher-level semantics which, in domain-specific applications, could be inferred either from the visual information itself or from associated information (e.g. subtitles).

The object ontology is presented in Fig. 2, where the possible intermediate-level descriptors and descriptor values are shown. Each intermediate level de-

descriptor value is mapped to an appropriate range of values of the corresponding low-level, arithmetic descriptor. With the exception of color (e.g. “black”) and direction (e.g. “low→high”) descriptor values, the value ranges for every low-level descriptor are chosen so that the resulting intervals are equally populated. This is pursued so as to prevent an intermediate-level descriptor value from being associated with a plurality of spatiotemporal objects in the database, since this would render it useless in restricting a query to the potentially most relevant ones. Overlapping, up to a point, of adjacent value ranges, is used to introduce a degree of fuzziness to the descriptor values; for example, both “slightly oblong” and “moderately oblong” values may be used to describe a single object.

Regarding color, a correspondence between the 11 basic colors [10] used as color descriptor values and the values of the HSV color space is heuristically defined. More accurate correspondences based on psychovisual findings are possible; this is however beyond the scope of this work. Regarding the direction of motion, the mapping between values for the descriptors “x direction”, “y direction” and the MPEG-7 *Motion Trajectory* descriptor is based on the sign of the cumulative displacement of the foreground spatiotemporal objects.

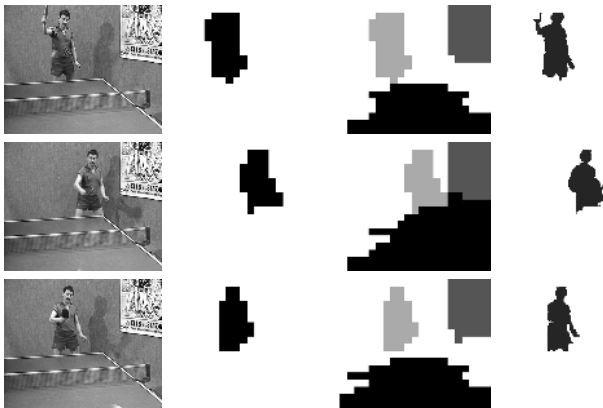
### 3.4 Relevance Feedback

After using the intermediate-level descriptors to narrow down the search to a set of potentially relevant spatiotemporal objects, relevance feedback is employed to produce a qualitative evaluation of the degree of relevance of each spatiotemporal object. The employed mechanism is based on a method proposed in [11], where it is used for image retrieval using global image properties under the query-by-example scheme. This method combines support vector machines (SVM) with a constrained similarity measure (CSM). Under the proposed scheme, the SVMs classify spatiotemporal objects to relevant or non-relevant using their low-level descriptor vectors, while the CSM proposed in [11] is modified to subsequently assign to each spatiotemporal object classified as relevant the minimum of the Euclidean distances between it and all positive training samples.

## 4 Experimental Results

The proposed algorithms were tested on known test sequences, as well as a collection of video shots. Results of the real-time compressed-domain segmentation algorithm are presented for the “Table-tennis” sequence (Fig. 3). The proposed segmentation approach imposes little additional computational burden to the MPEG decoder: excluding any processes of it, the proposed algorithm requires on the average 5.02 msec per processed CIF-format I/P-frame on an 800Mhz Pentium III.

For each video object created by applying the segmentation algorithm to a collection of video shots, MPEG-7 low-level descriptors were calculated and the mapping between them and the intermediate-level descriptors defined by the object ontology was performed. Subsequently, the object ontology was used



**Fig. 3.** Results of moving-object detection, final mask after background segmentation, and moving objects after pixel-domain boundary refinement using a Bayes classifier to reclassify specific pixels in a fashion similar to that of [12], for “Table-tennis”.

Query results for “red\_car”: shots 1 to 15 of 130



(a)

Query results for “red\_car”: shots 1 to 15 of 130



(b)

**Fig. 4.** Results for a “red car” query: (a) shots containing potentially relevant objects, identified using the intermediate-level descriptors, (b) results after one round of relevance feedback.

to define, using the available intermediate-level descriptors, semantic objects. Querying using these definitions resulted in initial results produced by excluding the majority of spatiotemporal objects in the database. Finally, one or more pages of potentially relevant spatiotemporal objects were presented to the user for manual evaluation and training of the SVM-based feedback mechanism. Results of this procedure for a “red car” query are presented in Fig. 4.



## 5 Conclusions

An algorithm for compressed video segmentation was presented in this paper, along with an indexing and retrieval scheme. Due to its real-time, unsupervised operation, the proposed algorithm is very suitable for content-based multimedia applications requiring the manipulation of large volumes of visual data. The proposed video indexing and retrieval scheme, based on the combination of the proposed segmentation algorithm with ontologies and relevance feedback, enabled the formulation of descriptive queries and allowed efficient retrieval of video segments.

## References

1. Al-Khatib, W., Day, Y., Ghafoor, A., Berra, P.: Semantic modeling and knowledge representation in multimedia databases. *IEEE Trans. on Knowledge and Data Engineering* **11** (1999) 64–80
2. O'Connor, N., Sav, S., Adamek, T., Mezaris, V., Kompatsiaris, I., Lui, T., Izquierdo, E., Bennstrom, C., Casas, J.: Region and Object Segmentation Algorithms in the Qimera Segmentation Platform. In: *Proc. Third Int. Workshop on Content-Based Multimedia Indexing (CBMI03)*. (2003)
3. Meng, J., Chang, S.F.: Tools for Compressed-Domain Video Indexing and Editing. In: *Proc. SPIE Conf. on Storage and Retrieval for Still Image and Video Databases IV*, Ishwar K. Sethi; Ramesh C. Jain; Eds. Volume 2670. (1996) 180–191
4. Sahouria, E., Zakhori, A.: Motion Indexing of Video. In: *Proc. IEEE Int. Conf. on Image Processing (ICIP97)*, Santa Barbara, CA (1997)
5. Babu, R., Ramakrishnan, K.: Compressed domain motion segmentation for video object extraction. In: *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*. Volume 4. (2002) 3788–3791
6. Mezaris, V., Kompatsiaris, I., Strintzis, M.G.: An Ontology Approach to Object-based Image Retrieval. In: *Proc. IEEE Int. Conf. on Image Processing (ICIP03)*, Barcelona, Spain (2003)
7. Yu, T., Zhang, Y.: Retrieval of video clips using global motion information. *Electronics Letters* **37** (2001) 893–895
8. Favalli, L., Mecocci, A., Moschetti, F.: Object tracking for retrieval applications in MPEG-2. *IEEE Trans. on Circuits and Systems for Video Technology* **10** (2000) 427–432
9. Sikora, T.: The MPEG-7 Visual standard for content description - an overview. *IEEE Trans. on Circuits and Systems for Video Technology*, special issue on MPEG-7 **11** (2001) 696–702
10. Berlin, B., Kay, P.: Basic color terms: their universality and evolution. Berkeley, University of California (1969)
11. Guo, G.D., Jain, A., Ma, W.Y., Zhang, H.J.: Learning similarity measure for natural image retrieval with relevance feedback. *IEEE Trans. on Neural Networks* **13** (2002) 811–820
12. Mezaris, V., Kompatsiaris, I., Strintzis, M.: A framework for the efficient segmentation of large-format color images. In: *Proc. IEEE Int. Conf. on Image Processing (ICIP02)*. Volume 1. (2002) 761–764

# Interoperability Support for Ontology-Based Video Retrieval Applications

Chrisa Tsinaraki, Panagiotis Polydoros, and Stavros Christodoulakis

Technical University of Crete, Laboratory of Distributed Multimedia Information Systems and Applications (TUC/MUSIC), University Campus, 73100 Kounoupidiana, Crete, Greece  
{chrisa, panpolyd, stavros}@ced.tuc.gr

**Abstract.** Domain ontologies are very useful for indexing, query specification, retrieval and filtering, user interfaces, even information extraction from audiovisual material. The dominant emerging language standard for the description of domain ontologies is OWL. We describe here a methodology and software that we have developed for the interoperability of OWL with the complete MPEG-7 MDS so that domain ontologies described in OWL can be transparently integrated with the MPEG-7 MDS metadata. This allows applications that recognize and use the MPEG-7 MDS constructs to make use of domain ontologies for applications like indexing, retrieval, filtering etc. resulting in more effective user retrieval and interaction with audiovisual material.

## 1 Introduction

The advent of the Internet and the digital multimedia demonstrated the extreme importance of standards for the industry. While in closed environments bound by the organizational walls the organizations could be content with their own software and hardware, in open environments where contact with remote companies or users via Internet or satellite links is of great importance, interoperability through the use of industry standards has become crucial.

In the multimedia industry the MPEG standards have lead the industrial efforts. MPEG-7 [9] [7] is today a very well accepted standard for describing aspects of the multimedia content related to retrieval and filtering, like content structuring metadata, user filtering metadata, usage metadata, segmentation metadata etc. Future work in this area will have to be based on the existing MPEG-7 standard, extending it in appropriate ways.

Retrieval and filtering of audiovisual data is a very important but difficult subject of research for the academia and the industry, and has received a lot of attention in scientific publications [12] [13] [14] [15] [16] [17]. It has been shown in many real-world applications that the retrieval effectiveness (as measured by the precision-recall curves for example) can be greatly improved when domain knowledge encoded in domain ontologies can be used for indexing and retrieval purposes. Since this is also true for audiovisual data, we have developed a methodology for extending the MPEG-7 content metadata with domain knowledge so that we improve the indexing and

retrieval of audiovisual content [17]. The extension of MPEG-7 content metadata with domain ontologies is done in a way that is transparent to the applications that only understand MPEG-7 so that they can still be operational, and also take advantage of the domain-specific extensions.

Domain ontologies are often described in domain ontology languages that allow rich ontology structures. OWL [4] is the dominant effort in the standardization of ontology languages and it is expected that both, many domain ontologies will exist in the future described in OWL, as well as that many scientists will be familiar with OWL and will be using it for the definition of new ontologies. It is therefore very important for the audiovisual industry to have a methodology for the interoperability of OWL with MPEG-7 and for the integration of domain ontologies expressed in OWL with MPEG-7.

In this paper we present a methodology and a software implementation that achieves the interoperability of the complete MPEG-7 MDS (including content metadata, filtering metadata etc.) with OWL. We also demonstrate how domain ontologies described in OWL can be integrated in the MPEG-7 MDS in a way that is transparent to the applications that understand MPEG-7. Finally we demonstrate how the domain ontologies that are integrated in various parts of the MPEG-7 MDS can be used to increase the retrieval effectiveness of queries, as well as the retrieval effectiveness of the filtering process using the MPEG-7 user profiles.

The work described in this paper is in the context of the DS-MIRF framework for semantic retrieval of audiovisual metadata, and extends our previous work described in [13] [16] [17] to cover all the aspects of the MPEG-7 MDS. Little has been published in the past in this area of research although the importance of domain ontologies in content recognition, indexing and retrieval is widely recognized [1] [2] [6] [10] [11] [12]. The work that is closest to ours is [5], where the RDF [8] ontology definition language is used to partially describe the MPEG-7 content metadata structures, but not the complete MPEG-7 MDS. This work [5] does not propose a specific methodology and/or software for the integration of domain-specific ontologies in MPEG-7.

Our approach is based on the definition of an OWL Upper Ontology, which fully captures the MPEG-7 MDS. The Upper Ontology is the basis for interoperability between OWL and the MPEG-7 MDS. We also define a methodology for the definition of domain ontologies based on the Upper Ontology. Finally we defined a set of transformation rules that map the domain ontologies that have been described in OWL to the MPEG-7 MDS in a way transparent to the applications of the MPEG-7 MDS.

The rest of the paper is organized as follows: The Upper Ontology capturing the MPEG-7 MDS is presented in section 2, while the applicability of our approach and the benefits of its usage in MPEG-7 applications are presented in section 3. Conclusions and future work are discussed in section 4.

## 2 An Upper Ontology Capturing the MPEG-7 MDS

Our approach for interoperability support in multimedia content service provision environments utilizes an ontology that captures the metadata model provided by the MPEG-7 MDS. This ontology, referred as the *Upper Ontology* in the rest of the paper,

has been implemented in OWL and is described in this section. We provide an overview of the MPEG-7 MDS in subsection 2.1. The methodology for the Upper Ontology definition is discussed in subsection 2.2.

## 2.1 Overview of the MPEG-7 MDS

We provide in this subsection a brief overview of the MPEG-7 MDS, which provides all the constructs needed for defining metadata that describe the multimedia content and the associated multimedia content services.

Each of the major components of the MPEG-7 MDS is composed of a set of *Description Schemes (DSs)*, essentially complex datatypes, used for the description of concepts in its scope. The MPEG-7 MDS is comprised of the following major components:

- *Basic Elements*, where the basic MDS elements are defined. Basic elements include schema tools (root element, top-level element and packages), basic datatypes, mathematical structures, linking and media localization tools as well as basic DSs, which are used as elementary components of more complex DSs.
- *Content Description & Management Elements*, which are used for the description of the content of a single multimedia document from several viewpoints. Information related to the content management is structured according to the *Creation & Production*, *Media* and *Usage* tools, while information related to the content description is structured according to the *Structural Aspects* and *Semantic Aspects* tools. These two sets of description mechanisms are interrelated.
- *Navigation & Access Elements*, where browsing is supported through multimedia content summary descriptions including information about possible variations of the content. Multimedia content variations can replace the original, if necessary, in order to adapt different multimedia presentations to the capabilities of the client terminals, network conditions, or user preferences.
- *Content Organization Elements*, where the organization of the multimedia content is addressed by classification, by modeling and by the definition of multimedia document collections.
- *User Interaction Elements*, which are used to describe user preferences regarding multimedia content, as well as material consumption aspects.

The MPEG-7 MDS has been defined by the standardization body using the MPEG-7 DDL, which is essentially based on the XML Schema Language [3], extended with the definition of the basic datatypes needed for the definition of the complex DSs of the MPEG-7 MDS.

## 2.2 Upper Ontology Definition Methodology

We describe in this subsection the methodology that we developed and applied for the definition of the OWL Upper ontology that fully captures the concepts of the MPEG-7 MDS. The Upper Ontology was defined according to the following methodological steps:

1. *MPEG-7 Simple Datatype Representation:* OWL does not provide mechanisms for simple datatype definition, but it permits the integration of simple datatypes defined in the XML Schema Language using the `rdfs:Datatype` construct. Thus, we store all the definitions of the simple datatypes of the MPEG-7 MDS in an XML schema file, represented by the `&datatypes;` XML entity. In addition, an `rdfs:Datatype` instance is defined in each of the ontology definition files for every simple datatype used in it. For example, in order to use the “zeroToOneType” datatype shown in Fig. 1, which represents real numbers between 0 and 1, we define the `rdfs:Datatype` instance of Fig. 2 in order to use the “zeroToOneType” type (defined in XML Schema) in the Upper Ontology.

```
<simpleType name="zeroToOneType">
  <restriction base="float">
    <minInclusive value="0.0"/>
    <maxInclusive value="1.0"/>
  </restriction>
</simpleType>
```

**Fig. 1.** Definition of the zeroToOneType datatype in the MPEG-7 MDS

```
<rdfs:Datatype rdf:about="&datatypes;zeroToOneType">
  <rdfs:isDefinedBy rdf:resource="&datatypes;" />
  <rdfs:label>zeroToOneType</rdfs:label>
</rdfs:Datatype>
```

**Fig. 2.** Definition of the `rdfs:Datatype` instance for the zeroToOneType datatype

Then, if there is a property of “zeroToOneType” type, which belongs to one of the Upper Ontology classes, the property type is denoted in the `rdfs:range` element of the property, as shown in Fig. 3.

```
<rdfs:range rdf:resource="&datatypes;zeroToOneType" />
```

**Fig. 3.** Definition of a property of zeroToOneType type

2. *MPEG-7 Complex Type Representation:* MPEG-7 complex types correspond to OWL classes, which represent groups of individuals that belong together because they share some properties. Thus, for every complex type defined in the MPEG-7 MDS we define a respective OWL class using the `owl:Class` construct, having as `rdf:ID` the value of the complex type name.
  - 2.1. *Simple Attribute Representation:* The simple attributes of the complex type of the MPEG-7 MDS are represented as OWL datatype properties, which relate class instances to datatype instances (e.g. integer, string etc.). Thus, for every simple attribute of the complex type a datatype property is defined using the `owl:DatatypeProperty` construct.
    - 2.1.1. The datatype property is “attached” to the OWL class through the `rdfs:domain` construct, which denotes the domain of the property. The value of the `rdfs:domain` of the datatype property is the `rdf:ID` value of the newly-defined class.
    - 2.1.2. The type of the values associated with the class through the datatype property is denoted in the `rdfs:range` element of the property. If the

attribute type is an enumerated type, the owl:DataRange construct is used in the context of rdfs:range.

2.2. *Complex Attribute Representation:* Complex attributes are represented as OWL object properties, which relate class instances. For every complex attribute of the complex type the following actions are performed:

2.2.1. An OWL class for the representation of the complex attribute instances is defined, if it does not already exist.

2.2.2. An OWL object property that relates the complex attribute instances with the complex type instances is defined using the owl:ObjectProperty construct. The domain and the range of the object properties are defined in the same way with the datatype properties.

2.3. *Subclassing:* For the representation of the subclass/superclass relationships holding for the complex type, the following actions are performed:

2.3.1. If the complex type is a subtype of another complex type, the subclass relationship is represented by an instance of the rdfs:subClassOf construct, which relates the newly-defined class its superclass.

2.3.2. If the complex type is a subtype of a simple type, a datatype property with rdf:ID “content” and rdfs:range of the simple type is associated with the newly-defined OWL class.

2.4. *Constraints:* Constraints regarding value, cardinality and type for simple and complex attributes are expressed using the owl:Restriction construct together with the owl:hasValue, owl:cardinality (owl:maxCardinality, owl:minCardinality and owl:FunctionalProperty may also be used) and owl:allValuesFrom constructs. Complex constraints are defined using the boolean operations owl:IntersectionOf, owl:UnionOf and owl:ComplementOf.

As an example, we show in Fig. 5 the definition of the OWL class “FilteringAndSearchPreferencesType” (subclass of the “DSType” that represents all the Descriptor Schemes), corresponding to the MDS complex type “FilteringAndSearchPreferencesType” shown in Fig. 4. The complex attribute “CreationPreferences” and the simple attribute “protected” are also shown in Fig. 4 and the corresponding “CreationPreferences” object property and the “protected” datatype property are shown in Fig. 5.

```
<complexType name="FilteringAndSearchPreferencesType">
  <complexContent>
    <extension base="mpeg7:DSType">
      <sequence>
        <element name="CreationPreferences"
type="CreationPreferencesType" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="protected" type="userChoiceType" use="optional"/>
    </complexContent>
  </complexType>
```

**Fig. 4.** The FilteringAndSearchPreferencesType MDS complex type

3. *MPEG-7 Relationship Representation:* Relationships between the OWL classes, which correspond to the complex MDS types, are represented by the instances of the “RelationBaseType” class and its subclasses. Every “RelationBaseType”

instance is associated with a source and a target metadata item through the homonym object properties.

The complete Upper Ontology has been designed using the above rules but is not shown here due to space limitations. It is an OWL-DL ontology, available at [18], which has been validated by the OWL species ontology validator<sup>1</sup>.

```
<owl:Class rdf:ID="FilteringAndSearchPreferencesType">
  <rdfs:subClassOf rdf:resource="#DSType" />
</owl:Class>
<owl:ObjectProperty rdf:ID="CreationPreferences">
  <rdfs:domain rdf:resource="#FilteringAndSearchPreferencesType" />
  <rdfs:range rdf:resource="#CreationPreferencesType" />
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:ID="protected">
  <rdfs:domain rdf:resource="#FilteringAndSearchPreferencesType" />
  <rdfs:range rdf:resource="#&datatypes;userChoiceType" />
  <rdf:type rdf:resource="#&owl;FunctionalProperty" />
</owl:DatatypeProperty>
```

**Fig. 5.** The FilteringAndSearchPreferencesType OWL class

### 3 MPEG-7 Application Support

We present in this section the use of our approach for metadata management in multimedia content service environments. Our approach includes, in addition to the Upper Ontology described in the previous section, the integration of OWL domain ontologies (lower ontologies) in order to provide higher quality content services. We focus here in the description of the advantages of our approach in search and filtering services.

In order to verify our design and implementation we have developed a complete domain ontology (lower ontology) for soccer games in OWL and have implemented the software needed for the transformation of OWL/RDF metadata defined using the Upper Ontology and the domain ontologies to MPEG-7 compliant metadata [16]. In addition, we have developed an MPEG-7 compliant API that supports ontology-based retrieval and filtering [13].

We discuss in the next subsections the methodology for domain-specific ontology definition and integration to the Upper Ontology (subsection 3.1) and the retrieval and filtering support provided (subsection 3.2).

#### 3.1 Methodology for the Integration of OWL Domain Ontologies

In this subsection we present the methodology for the definition and integration of domain ontologies that extend the semantics encapsulated in the Upper Ontology with domain knowledge.

<sup>1</sup> The OWL species validator, available at <http://phoebus.cs.man.ac.uk:9999/OWL/Validator>, validates OWL ontologies and checks if an ontology conforms to one of the OWL species.

The domain ontologies comprise the second layer of the semantic metadata model used in the DS-MIRF framework [14] [15], with the first layer of the model encapsulated in the Upper Ontology. Thus, the classes representing the domain-specific entities should be defined in a way that extends the Upper Ontology. Having these in mind, the domain ontologies are defined according to the following methodological steps:

- 1 Domain-specific entity types are represented by OWL classes that are subclasses of the appropriate Upper Ontology classes. For example, in a football tournament application the “FootballTeam” subclass of the “OrganizationType” Upper Ontology class, is used for the representation of football teams as is shown in Fig. 6.

```
<owl:Class rdf:ID="FootballTeam">
  <rdfs:subClassOf rdf:resource="#OrganizationType" />
</owl:Class>
```

**Fig. 6.** OWL Definition the FootballTeam class

- 1.1 Attributes (both simple and complex) not present in the superclass are represented as appropriate object or datatype properties.
- 1.2 Additional constraints may be applied on the attributes inherited from the parent class, in order to guide the indexers to produce valid metadata.
- 2 Relationships with additional restrictions compared with the ones of the general relationships defined in the Upper Ontology are usually needed (e.g. a Goal event may be related to player instances as goal agents). In these cases, appropriate subclasses of “RelationBaseType” or of its appropriate subclass are defined and all the restrictions needed are applied to the newly defined classes.

A more detailed discussion on this methodology and its application for the description of the semantics of soccer games can be found in [16].

### 3.2 Search and Filtering Support

We present in this subsection the advantages of our approach in search and filtering services. The retrieval and filtering support is based on the query API developed in the context of the DS-MIRF framework [13].

The end-users may now pose semantic queries on the semantics of the audiovisual content using transparently the API functions through the appropriate interface. The queries may be based on the general constructs provided by MPEG-7 (queries 1, 5 and 6 of Table 1) or on the domain knowledge (queries 2, 3 and 4). Such queries are shown in Table 1, where we assume that the ID of the person named “Ronaldo” is P1 and the ID of the person named “Kahn” is P2. We also assume that the ID of the soccer stadium named “Old Trafford” is SP1, the ID of Europe is SP2 and the ID of the date 1/1/2003 is ST1. It is obvious from the examples that the queries that make use of the domain knowledge are more expressive than the more general ones and their results will be better in terms of precision/recall.



**Table 1.** Semantic Query Examples

Query	Description in Free Text
1. GetSegment(P1 Person null)	"Give me the segments where Ronaldo appears" (not only as a player!)
2. GetSegmentMQT(null Event Goal AND P1 Person Player hasCauserOf)	"Give me the segments where the player Ronaldo scores"
3. GetSegmentMQT(null Event Goal AND (P1 Person Player hasCauserOf) (P2 Person Player hasPatientOf))	"Give me the segments where the player Ronaldo scores against the player Kahn"
4. GetSegmentMQT(null Event Goal AND (ST1 SemanticTime GameTime hasTimeOf) (SP1 SemanticPlace SoccerStadium hasPlaceOf))	"Give me the segments where a goal takes place in 1/1/2003 in the soccer stadium Old Trafford"
5. GetSegmentMQT(null SemanticTime null AND ST1 SemanticTime null after)	"Give me the segments referring to time after 1/1/2003"
6. GetSegmentMQT(null SemanticPlace null AND SP2 SemanticPlace null inside)	"Give me the segments where places inside Europe appear"

Our methodology can be used also for enhancing the user profiles with domain-specific filtering preference definitions. Consider now a user who wants to denote in his preference profile that he is interested in watching the extra time of soccer games. This can be achieved when he sets his preference conditions regarding soccer games. If domain knowledge has not been encapsulated in the application he uses, he can approximately specify the time point that the extra time (overtime) begins and the corresponding end time point (relative to the game start). In this case, if we assume that the ID of the approximate start time point is STP1 and the ID of the approximate end time point is STP2, the API query used (transparently to him) for the retrieval of the corresponding video segment is the query shown in Fig. 7. It is obvious that the audiovisual segment returned to him may contain events before or after the extra time and that not all the extra time duration may be retrieved (e.g. because of a delay) if there was extra time given for the game. If there was no extra time given, an audiovisual segment with irrelevant content should be returned.

```
GetSegmentMQT(null SemanticTime null AND STP1 SemanticTime null after
AND STP2 SemanticTime null before)
```

**Fig. 7.** Query for the approximate retrieval of the segment containing the extra time of a soccer game

If there exists domain knowledge, only if there was extra time given for the game the appropriate segment will be returned. The API query used, transparently to the user, is the one shown in Fig. 8.

```
GetSegment(null SemanticTime ExtraTime)
```

**Fig. 8.** Query for the retrieval of the segment containing the extra time of a soccer game

## 4 Conclusions – Future Work

In this paper we have presented a methodology for interoperability support between MPEG-7 and OWL, based on an OWL Upper Ontology that fully captures the semantics of the MPEG-7 MDS. The integration of domain specific knowledge in multimedia content applications is done through the extension of the Upper Ontology with OWL domain ontologies (lower ontologies). We have described the Upper Ontology definition methodology, as well as a methodology for the definition of domain ontologies that extend the Upper Ontology, in order to fully describe the concepts of specific application domains in a manner transparent to MPEG-7 applications. The complete OWL Upper Ontology describing the MPEG-7 MDS is available in [18]. In addition, we have presented the use of our approach in search and filtering services. The usage of the approach was presented through its application in MPEG-7-based multimedia content services.

Our future research in the area includes:

- The complete development of the *MOREL (Multimedia, Ontology-based REtrieval Language)* language based on the existing query API. The MOREL language aims to support queries that make use of ontologies in multimedia content service environments.
- The application of our approach (utilization of the Upper Ontology and of domain ontologies extending it) in applications from all aspects covered by the MPEG-7 MDS (user preferences, summarization etc.) in addition to applications for the semantic description of audiovisual content.

**Acknowledgments.** The work presented in this paper was partially funded in the scope of the DELOS II Network of Excellence in Digital Libraries (IST – Project Record Number 26059).

## References

1. J. Assfalg, M. Bertini, C. Colombo, A. Del Bimbo, "Semantic Annotation of Sports Videos", *IEEE MultiMedia* 9(2): 52-60 (2002)
2. M. Doerr, J. Hunter, C. Lagoze, "Towards a Core Ontology for Information Integration", *Journal of Digital Information*, Volume 4 Issue 1, April 2003
3. D. Fallside, "XML Schema Part 0: Primer", W3C Recommendation, 2001, <http://www.w3.org/TR/xmlschema-0/>
4. D. Mc Guinness, F. van Harmelen, "OWL Web Ontology Language Overview", W3C Candidate Recommendation, 2003, <http://www.w3.org/TR/owl-features/>
5. J. Hunter, "Adding Multimedia to the Semantic Web - Building an MPEG-7 Ontology", *International Semantic Web Working Symposium (SWWS)*, Stanford, July 30 - August 1, 2001
6. J. Hunter, "Enhancing the Semantic Interoperability of Multimedia through a Core Ontology", *IEEE Transactions on Circuits and Systems for Video Technology*, Special Issue on Conceptual and Dynamical Aspects of Multimedia Content Description, Feb 2003

7. ISO/IEC JTC 1/SC 29/WG 11/N3966, "Text of 15938-5 FCD Information Technology – Multimedia Content Description Interface – Part 5 Multimedia Description Schemes", Singapore, 2001
8. G. Klyne, J. Carroll, "Resource Description Framework (RDF): Concepts and Abstract Syntax", W3C Working Draft, 2003, <http://www.w3.org/TR/rdf-concepts/>
9. MPEG Group, "MPEG-7 (Multimedia Content Description Interface)", <http://www.chiariglione.org/mpeg/index.htm>
10. M. Naphade, J. Smith: "A Hybrid Framework for Detecting the Semantics of Concepts and Context", in Proceedings of CIVR 2003, Urbana, IL, July 24-25, 2003, pp 196-205
11. HJ Nock, G Iyengar, C Neti, "Speaker Localisation using Audio-Visual Synchrony: An Empirical Study", in Proceedings of CIVR 2003, Urbana, IL, July 24-25, 2003, pp 488-499
12. R. Troncy, "Integrating Structure and Semantics into Audio-visual Documents", 2nd International Semantic Web Conference (ISWC), 20-23 October 2003, Sanibel Island, Florida, USA
13. C. Tsinaraki, E. Fatourou, S. Christodoulakis, "An Ontology-Driven Framework for the Management of Semantic Metadata describing Audiovisual Information", in Proc. of CAiSE, Velden, Austria, 2003, pp 340-356
14. C. Tsinaraki, S. Papadomanolakis, S. Christodoulakis, "A Video Metadata Model supporting Personalization & Recommendation in Video-based Services", in Proc. of MDDE Workshop (in conjunction with RETIS), Lyon, France, 2001, pp. 104-109
15. C. Tsinaraki, S. Papadomanolakis, S. Christodoulakis, "Towards a two - layered Video Metadata Model", in Proc. of DEXA Workshop - DLib, Munich, Germany, 2001, pp 937-941
16. C. Tsinaraki, P. Polydoros, S. Christodoulakis, "Integration of OWL ontologies in MPEG-7 and TVAnytime compliant Semantic Indexing", in Proc. of CaiSE 2004
17. C. Tsinaraki, P. Polydoros, F. Kazasis, S. Christodoulakis, "Ontology-based Semantic Indexing for MPEG-7 and TV-Anytime Audiovisual Content", Special issue of Multimedia Tools and Applications Journal on Video Segmentation for Semantic Annotation and Transcoding, 2004 (accepted for publication)
18. C. Tsinaraki, P. Polydoros, S. Christodoulakis, "Interoperability of OWL with the MPEG-7 MDS", Technical Report, Technical University of Crete / Laboratory of Distributed Multimedia Information Systems and Applications (TUC/MUSIC), <http://www.music.tuc.gr/TR/OWL-MPEG7.zip>

# A Test-Bed for Region-Based Image Retrieval Using Multiple Segmentation Algorithms and the MPEG-7 eXperimentation Model: The Schema Reference System\*

Vasileios Mezaris<sup>1,2</sup>, Haralambos Doulaverakis<sup>2</sup>,  
Raul Medina Beltran de Otalora<sup>3</sup>, Stephan Herrmann<sup>3</sup>, Ioannis Kompatsiaris<sup>2</sup>,  
and Michael G. Strintzis<sup>1,2</sup>

<sup>1</sup> Information Processing Laboratory, Electrical and Computer Engineering  
Department, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece

<sup>2</sup> Informatics and Telematics Institute (ITI)/ Centre for Research and Technology  
Hellas (CERTH), Thessaloniki 57001, Greece

<sup>3</sup> Institute for Integrated Systems, Munich University of Technology, Munich  
D-80290, Germany

**Abstract.** The aim of the SCHEMA Network of Excellence is to bring together a critical mass of universities, research centers, industrial partners and end users, in order to design a reference system for content-based semantic scene analysis, interpretation and understanding. In this paper, recent advances in the development of the SCHEMA reference system are reported, focusing on the application of region-based image retrieval using automatic segmentation. More specifically, the first and the second version of the reference system are presented and the motivation behind the different approaches followed during the development of these two versions is discussed. Experimental results for both systems, using a common collection of images, are shown. Additionally, a comparative evaluation of the two versions both in terms of retrieval accuracy and in terms of time-efficiency is performed, allowing the evaluation of the system as a whole as well as the evaluation of the usability of different components integrated with the reference system, such as the MPEG-7 eXperimentation Model. This illustrates the suitability of the SCHEMA reference system in serving as a test-bed for evaluating and comparing different algorithms and approaches pertaining to the content-based and semantic manipulation of visual information, ranging from segmentation algorithms to indexing features and methodologies.

## 1 Introduction

As a result of the continuously accelerated generation and distribution of digital media and in particular still images, the efficient access and retrieval of visual

---

\* This work was supported by the EU project SCHEMA “Network of Excellence in Content-Based Semantic Scene Analysis and Information Retrieval”, [www.schema-ist.org](http://www.schema-ist.org) (IST-2001-32795).

information has emerged in recent years as an important research direction. Many approaches to image retrieval have appeared in the literature, ranging from content-based ones to approaches exploiting other modalities such as text, while at the same time the importance of the media retrieval task has also motivated the introduction of the relevant MPEG-7 International Standard [1]. The latter is formally named “Multimedia Content Description Interface”.

Although initial attempts for image retrieval were based on exploiting text (e.g. image captions), the Query-by-Example paradigm was soon established as the prevalent methodology for addressing the problem of image retrieval from generic collections. The Query-by-Example paradigm has been explored in conjunction with both *coarse granularity* and *fine granularity* preprocessing of the image data, where the latter signifies the analysis of the image to meaningful regions while the former involves the processing of images as a whole [2]. With recent works documenting the superiority of fine granularity approaches over coarse granularity ones [3], the two most important unknowns in the process of building a content-based image retrieval system are the method to be used for segmenting the images to meaningful regions and the features that should be extracted for these regions and employed for matching. These issues are highly dependent upon the image data to be used, thus no universal solution exists.

To facilitate the evaluation of the suitability of different such tools for a given image retrieval application, this paper presents the Schema Reference System. This is a content-based image retrieval system that employs multiple segmentation algorithms and indexing and retrieval subsystems, thus being able to serve as a test-bed for comparisons between them. The Schema Reference System also allows the easy integration of additional algorithms and tools with it and their subsequent comparative evaluation.

The paper is organized as follows: in section 2 the first version of the SCHEMA reference system, introducing the use of four segmentation algorithms, is presented. In section 3 the development of the second version is discussed, emphasizing on the integration of the reference system with the MPEG-7 eXperimentation Model. Section 4 contains experimental evaluation and comparison of the developed methods, and finally, conclusions are drawn in section 5.

## 2 Reference System v.1.0

### 2.1 Visual Medium Analysis

As already mentioned, the SCHEMA Reference System [4] has adopted a fine granularity approach to image indexing and retrieval, thus requiring the use of a segmentation algorithm for decomposing the images to meaningful regions. The use of a segmentation algorithm for region-based image retrieval has several advantages, mainly deriving from the fact that the user of an image retrieval system typically queries for objects similar to one such depicted in a key-image, rather than simply for similar images. Thus, using image segments (regions) that are closer to the notion of objects than the entire images themselves can

significantly improve the accuracy of the retrieval process. The imperfection of any segmentation algorithm is, however, a limiting factor in such schemes. To counter this drawback and to provide at the same time a test-bed for the comparison of different segmentation algorithms in terms of their suitability for the application of content-based image retrieval, a number of different segmentation algorithms have been integrated with the SCHEMA reference system. The different segmentation masks produced by these algorithms for a given image are simultaneously presented to the user, to allow for the one most suited to the user needs at the given time to be employed for initiating the query.

There have been four segmentation algorithms integrated so far with the reference system. All were previously integrated in the Qimera framework [5, 6], which provides common input/output formats, thus facilitating their rapid subsequent integration with the reference system. These algorithms are the following:

- Pseudo Flat Zone Loop algorithm (PFZL), contributed by Munich University of Technology - Institute for Integrated Systems.
- Modified Recursive Shortest Spanning Tree algorithm (MRSST), contributed by Dublin City University
- K-Means-with-Connectivity-Constraint algorithm (KMCC), contributed by the Informatics and Telematics Institute / Centre for Research and Technology - Hellas.
- Expectation Maximization algorithm (EM) in a 6D colour/texture space, contributed by Queen Mary University of London.

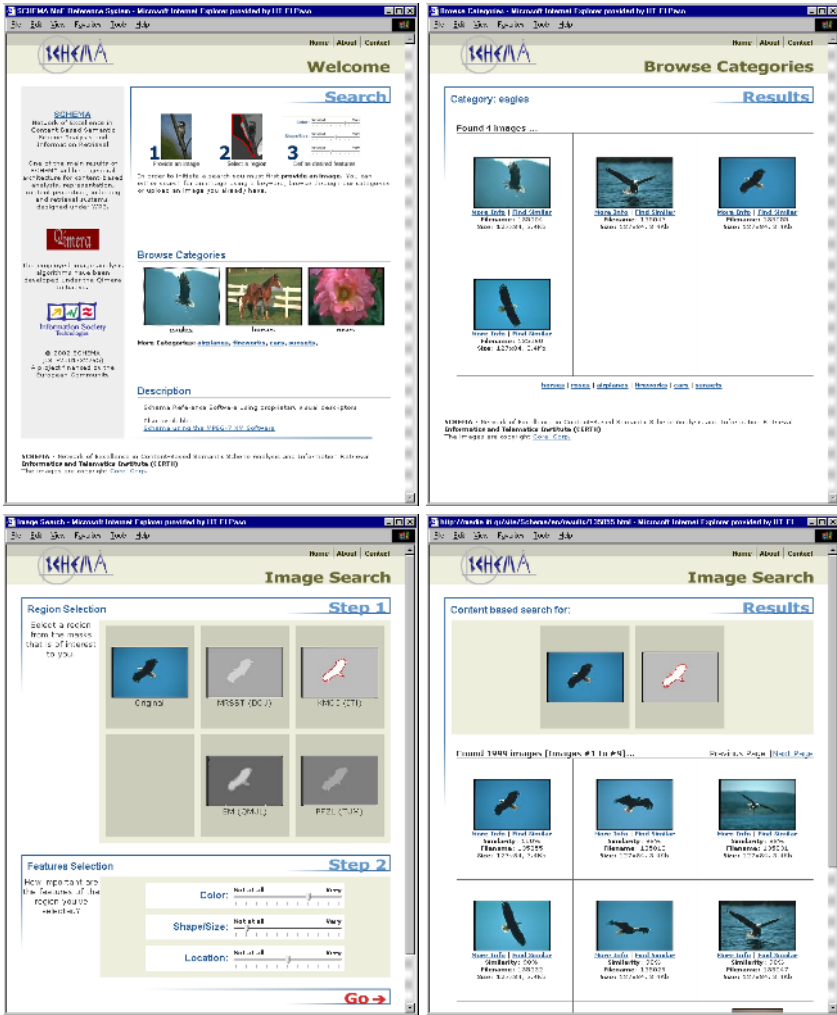
A more detailed description and references to these algorithms can be found in [7].

## 2.2 Graphical User Interface Functionality

The Graphical User Interface of the Reference System is built around the Query-by-Example paradigm. Thus, the user is first presented with several image categories (Fig. 1(a)); after selecting a category, a few images belonging to it are presented to the user for the purpose of initiating the query (Fig. 1(b)). The user selects an image and is subsequently presented with the segmentation masks generated for this image by the integrated segmentation algorithms (Fig. 1(c)). After selecting a region of the image, using any of the available segmentations, similarity search is performed in the image collection and its results are presented (Fig. 1(d)). Any of the result images (Fig. 1(d)) can then be employed for initiating a new query using the corresponding segmentations, as in (Fig. 1(c)).

## 2.3 Indexing Features and Matching

The region-based indexing features used in the first version of the Schema Reference System are non-standardized descriptors, capturing the color, position, size and shape properties of each region. More specifically, the employed features are:



**Fig. 1.** Graphical User Interface of the Schema Reference System: (a) category selection, (b) query image selection, (c) query region selection using any segmentation algorithm, (d) presentation of query results, which can be used for initiating a new query.

- Linearized color histograms in the RGB color space, quantized to 8 bins per color component and normalized so that the sum of the bin values for each color component equals to 1, thus resulting in 24 color features.
- Coordinates of the center of gravity of the region in the image grid, normalized by division with the image dimensions in pixels (2 position features).

- Size of the region, expressed as the ratio of the number of pixels assigned to it over the total number of pixels of the image.
- Normalized eccentricity, calculated using the covariance matrix of the region and Principal Component Analysis.

Matching of regions is performed using a Euclidean distance and different weights for the features belonging to the three different feature categories (color, position and size/shape). The employed weights can be adjusted by the user, as shown in Fig. 1(c).

### 3 Reference System v.2.0

#### 3.1 Relation to v.1.0

The second version of the Schema Reference System, also known as SchemaXM [8], addresses the exact same problem: the content-based indexing and retrieval of still color images using the query-by-example paradigm. The same *fine granularity* approach to visual information indexing and retrieval has been adopted and is being supported by the four aforementioned segmentation algorithms. Consequently, from a user's point of view the process of querying has not changed and thus, the user interface has remained almost unaltered.

The main difference between the two implementations lies in the indexing subsystem. Specifically, the non-standardized indexing features employed in the first version of the reference system have been replaced in this version with a collection of standardized MPEG-7 Visual Descriptors, which effectively characterize the color, texture and shape properties of each region. The underlying matching procedure has been changed accordingly, to make use of suitable Descriptor-specific similarity measures. The use of standardized MPEG-7 Descriptors and corresponding similarity measures has been made possible by integrating a variant of the MPEG-7 eXperimentation Model (XM) [9] with the Schema Reference System.

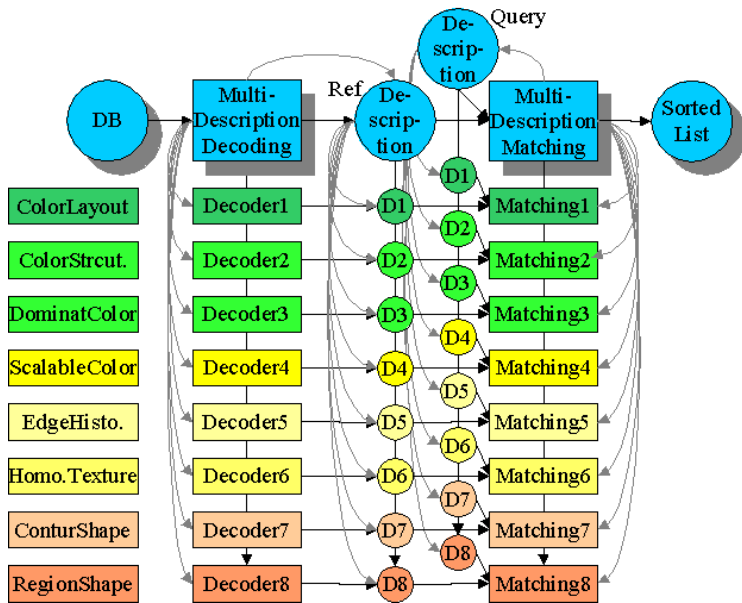
#### 3.2 Indexing and Retrieval Using MPEG-7 XM

The set of MPEG-7 descriptors that were selected for describing regions in the Schema reference system are summarized in table 1. These descriptors are instantiated using an abstract descriptor, the so called MultiDescriptor for images. This abstract descriptor module simply encapsulates the memory management for the selected descriptors and allows calling the extraction and the matching of all the descriptors as if they were parts of a single descriptor. Figure 2 shows the architecture of the matching process using the abstract descriptor (marked in blue). When instantiating the MultiDescriptor objects, the corresponding objects of the selected descriptors are also instantiated. This behavior is indicated by the grey (curved) arrows. When creating the processing chain, the descriptor modules are also connected to their processing chains (marked with black arrows).



**Table 1.** MPEG-7 Descriptors used

Color descriptors	Color Layout Color Structure Dominant Color Scalable Color
Texture descriptors	Edge Histogram Homogeneous Texture
Shape descriptors	Contour Shape Region Shape



**Fig. 2.** Architecture of the MultiDescriptor module, i.e. of the matching chain using multiple descriptors.

An essential problem when combining matching values of different visual descriptors is the fact that the distance functions are not normalized in any way. To alleviate this problem, the MultiDescriptor module performs a simple normalization of the working point of the matching functions. The working point is the value of the distance between two descriptions that, if not exceeded, signifies that the two descriptions should be treated as similar. If, on the opposite, the distance value exceeds this threshold, the two descriptions are assumed to be different. The working points for the individual descriptors were determined experimentally. Subsequent to normalization using the working points, i.e making the latter equal to 1.0 for all descriptors, the different distance functions can be

scaled in a linear way. Thus, in order to generate a single overall matching value for each region, a weighted linear combination of the individual distance values is simply calculated.

### 3.3 Implementation Issues

The original MPEG-7 eXperimentation Model is a simple command line program; for doing a similarity search using the selected visual descriptors, the program reads the descriptions for the entire image collection from the .mp7 files containing the descriptor bit stream. Additionally, it extracts the description of the image used for submitting the query (key-image) during query execution. As a result, for every search process the descriptions database is accessed and decoded and the key-image is processed, leading to unnecessary overheads in the query execution process. To accelerate the search procedure, a different approach is adopted in SchemaXM.

First, if the query image is already part of the database, the query description is not extracted again from the image data. This is achieved by restructuring the MPEG-7 XM search and retrieval application using the Visual XM Library and taking advantage of the modularity of the XM software.

Secondly, the MPEG-7 XM is used as a server that keeps running in the background, accepting new queries and delivering the corresponding retrieval results. This removes the need for continuously reading and decoding the MPEG-7 bitstreams, since these tasks can then be performed only once during server initiation and be omitted at query time. The use of the MPEG-7 XM as a server required the introduction of extensions to the XM software, an approach also investigated in [10].

Using the two aforementioned methods for integration with the MPEG-7 XM resulted in significant improvement of the time-efficiency of the SchemaXM system as compared to using the original MPEG-7 XM software for the purpose of integration, as discussed in the following section.

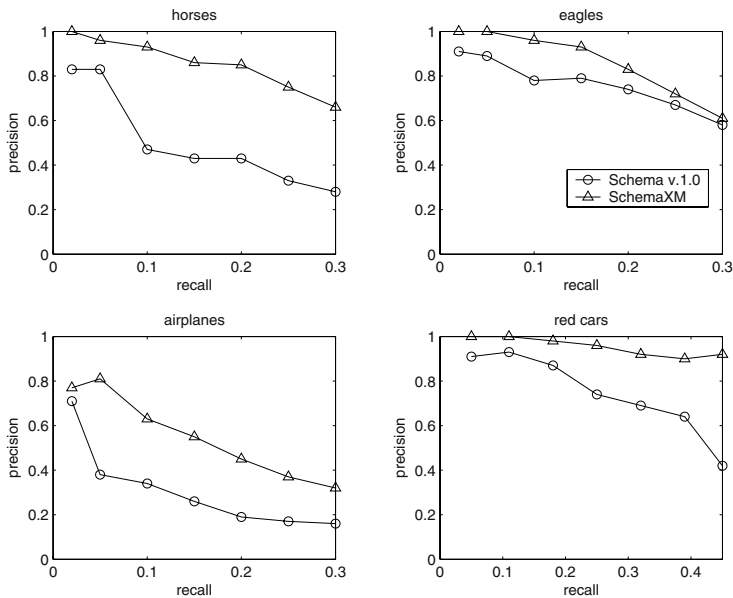
## 4 Experimental Results

Both versions of the Schema reference system were tested, for the purpose of performance evaluation, on a common collection of 2000 still images of the Corel gallery [11]. These images were pre-assigned to 20 categories (e.g. eagles, flowers, cars, etc.), each containing 100 images, while subcategories (e.g. red cars) were manually defined whenever necessary. This category / subcategory membership information was employed only for evaluating the performance of the reference system by calculating precision-recall diagrams for specific query categories. Note that the term *precision* is defined as the fraction of retrieved images which are relevant, and the term *recall* as the fraction of relevant images which are retrieved [3]. The aforementioned precision-recall curves were produced by averaging the results of five different queries for objects of the same category, to allow for objective results to be extracted, while for both systems the same segmentation

**Table 2.** Average query execution times

System	Time(sec)
Schema reference system v.1.0	< 1
MPEG-7 eXperimentation Model (original version)	15
SchemaXM (with MPEG-7 XM running as a server)	6

algorithm was employed when initiating a query using a specific image. These results are presented in Fig. 3, where it can be seen that the the second version of the reference system (SchemaXM), employing the MPEG-7 XM for indexing feature extraction and for matching, performs consistently better than the first version of the Schema reference system.



**Fig. 3.** Precision-Recall diagrams for comparing between the two versions of the Schema reference system.

However, this improvement is only achieved at the expense of the time required for query execution. As can be seen in table 2, the SchemaXM requires on the average 6 seconds to process a query. This is a significant improvement as compared to using the original version of MPEG-7 XM, achieved due to the use of the underlying MPEG-7 XM retrieval module as a server; however, SchemaXM still requires considerably more query execution time than the first version of the Schema reference system. It should be noted that such time-efficiency results are

to some extent expected, since the MPEG-7 XM MultiDescriptor search application employs a plethora of complex descriptors and correspondingly complex matching functions, as opposed to the simple proprietary descriptors and the Euclidean distance employed in the first version of the system.

## 5 Conclusions

Recent advances in the development of the SCHEMA reference system were reported in this paper. The two versions presented employ different retrieval modules, based on proprietary and MPEG-7 standardized descriptors respectively, while both make use of a range of segmentation algorithms. The variation of employed analysis tools and retrieval methodologies allows effective evaluation of their suitability for use in a content-based image retrieval system. This, along with the possibility of integrating additional such tools with the SCHEMA reference system, illustrates the potential use of it as a test-bed for evaluating and comparing different algorithms and approaches.

Future research will concentrate on further improving the time-efficiency of SchemaXM by implementing an indexing mechanism for the descriptor database, on enabling more extensive interactivity by means of relevance feedback, and on introducing the use of other modalities in combination with content-based features, in order to improve the retrieval accuracy.

## References

1. Chang, S.F., Sikora, T., Puri, A.: Overview of the MPEG-7 standard. *IEEE Trans. on Circuits and Systems for Video Technology* **11** (2001) 688–695
2. Al-Khatib, W., Day, Y., Ghafoor, A., Berra, P.: Semantic modeling and knowledge representation in multimedia databases. *IEEE Trans. on Knowledge and Data Engineering* **11** (1999) 64–80
3. Carson, C., Belongie, S., Greenspan, H., Malik, J.: Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **24** (2002)
4. Schema reference system v.1.0: (<http://media.iti.gr/site/Schema>)
5. O'Connor, N., Adamek, T., Sav, S., Murphy, N., Marlow, S.: QIMERA: A Software Platform for Video Object Segmentation and Tracking. In: *Proc. Workshop on Image Analysis For Multimedia Interactive Services*, London, UK (2003)
6. O'Connor, N., Sav, S., Adamek, T., Mezaris, V., Kompatsiaris, I., Lui, T., Izquierdo, E., Bennstrom, C., Casas, J.: Region and Object Segmentation Algorithms in the Qimera Segmentation Platform. In: *Proc. Third Int. Workshop on Content-Based Multimedia Indexing (CBMI03)*. (2003)
7. Schema Deliverable D3.1: (<http://www.schema-ist.org/SCHEMA/files/document/30-03-2004/D3.1.pdf>)
8. Schema reference system v.2.0: (<http://media.iti.gr/site/SchemaXM>)
9. MPEG-7 XM software: ([http://www.lis.ei.tum.de/research/bv/topics/mmdb/e\\_mpeg7.html](http://www.lis.ei.tum.de/research/bv/topics/mmdb/e_mpeg7.html))
10. BUSMAN IST Project: (<http://busman.elec.qmul.ac.uk/>)
11. Corel stock photo library: (Corel Corp., Ontario, Canada)

# ICBR – Multimedia Management System for Intelligent Content Based Retrieval

Janko Čalić<sup>1</sup>, Neill Campbell<sup>1</sup>, Majid Mirmehdi<sup>1</sup>, Barry T. Thomas<sup>1</sup>  
Ron Laborde<sup>2</sup>, Sarah Porter<sup>2</sup>, and Nishan Canagarajah<sup>2</sup>

<sup>1</sup> Department of Computer Science,

<sup>2</sup> Department of Electrical & Electronic Engineering,

University of Bristol, Merchant Venturers Building,

Woodland Road, Bristol BS8 1UB, UK

{janko.calic, neill.campbell, m.mirmehdi, barry.thomas,  
ron.laborde, sarah.porter, nishan.canagarajah}@bristol.ac.uk

**Abstract.** This paper presents a system designed for the management of multimedia databases that embarks upon the problem of efficient media processing and representation for automatic semantic classification and modelling. Its objectives are founded on the integration of a large-scale wildlife digital media archive with a manually annotated semantic metadata organised in a structured taxonomy and media classification system. Novel techniques will be applied to temporal analysis, intelligent key-frame extraction, animal gait analysis, semantic modelling and audio classification. The system demonstrator will be developed as a part of an ICBR project within the 3C Research programme of convergent technology research for digital media processing and communications.

## 1 Introduction

The overwhelming growth of multimedia information in private and commercial databases, as well as its ubiquity throughout the World Wide Web, present new research challenges in computing, data storage, retrieval and multimedia communications. Having in mind the user's need to intuitively handle this vast multimedia information the development of content based multimedia indexing and retrieval appears to be in the spotlight of multimedia and computer vision research. However, evolution of a functional multimedia management system is hindered by the "semantic gap"; a discontinuity between the simplicity of content descriptions that can be currently computed automatically and the richness of semantics in user's queries posed for media search and retrieval [1]. In order to bridge that gap, it is essential to focus our research activities towards the knowledge behind the links that connect perceptual features of the analysed multimedia and their meaning. Thus, a large-scale system that merges the semantic text-based retrieval approach to multimedia databases with content-based feature analysis and investigates the signification links between them ought to be the next milestone of research in the multimedia management field.

The ICBR (Intelligent Content-based Retrieval) project aims at developing a large-scale centralized multimedia server in order to enable large quantities of media to be stored, indexed, searched, retrieved, transformed and transmitted within a framework encompassing issues related to multimedia management, content analysis and semantic multimedia retrieval. The system is designed to be large enough to test the problems associated with real-word demands for such media content with regard to bandwidth, processing power, storage capacity and user requirements. Moreover, ICBR brings a unique opportunity to tackle semantic gap issues by integrating a large multimedia database together with the media's semantic description organised in a structured taxonomy.

## 2 Related Work

In the first generation of visual retrieval systems attributes of visual data were extracted manually, entailing a high level of content abstraction. Though operating on a conceptual level, search engines worked only in the textual domain and the cost of annotation was typically very high.

Second-generation systems addressed perceptual features like colour, textures, shape, spatial relationships obtaining fully automated numeric descriptors from objective measurements of the visual content and supporting retrieval by content based on combinations of these features. A key problem with second-generation retrieval systems was the semantic gap between the system and users. Query by Example retrieval is a typical technique that utilises this paradigm [2], [3].

However, retrieval of multimedia is generally meaningful only if performed at high levels of representation based on semantically meaningful categories. Furthermore, human visual cognition is much more concerned with the narrative and discourse structure of media than merely with its perceptual elements [4]. Therefore, a lot of research effort has been put recently into developing a system that will enable automatic semantic analysis and annotation of multimedia, video as the most complex media in particular.

Initially, temporal structure of media is analysed by tracking spatial domain sequence features [5], or more recently compressed domain features [6]. A video sequence is summarised by choosing a single key-frame [7] or a structured set of key-frames [8] to represent the content in a best possible way. Low-level features like colour, texture, shape, etc. are extracted and stored in the database as video metadata. Using this information, various methods of media representation have been proposed [9], [10] targeting user centred retrieval and the problem of semantic gap. Utilising metadata information, attempts to apply semantic analysis to a limited contextual space were presented [11].

In order to bring together research and industrial knowledge and develop large reference content-based retrieval systems essential for real evaluation and experimentation, a number of collaborative projects in the field of content-based multimedia retrieval have been set up: IST Network of Excellence SCHEMA [12], reference system COST211[13], digital video library INFOMEDIA [14], etc.

### 3 ICBR Approach

Incorporating a large manually annotated multimedia archive and its structured semantic classification ICBR has a unique potential to tackle a wide spectrum of research problems in content-based multimedia retrieval. The major guideline of ICBR development process is the semantic information, considered as the *ground truth*, upon which content-analysis algorithms base their internal representations and knowledge. Not only does the semantic annotation of a vast amount of multimedia data enable semantic inference, classification and recognition, but it enhances the feature extraction process as well. This approach imitates human semantic visual cognition where high-level semantic categories influence the way low-level features are perceived by the visual system.

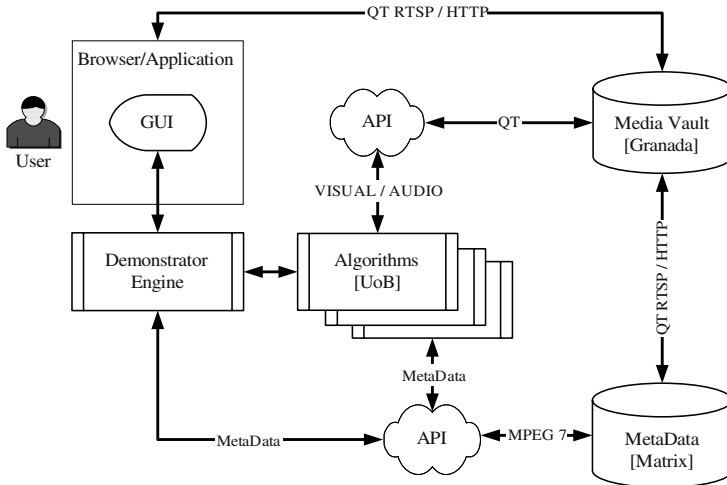
In the initial stage of temporal analysis and content summarisation, production knowledge and ground-truth annotation are utilised in algorithm design to achieve robust results on real-world data. Production information like camera artefacts, overexposures, camera work as well as the shot composition type and region spatio-temporal relations are automatically extracted to enable content summarisation and temporal representation on higher a semantic level than conventional summarisation methods.

In addition to algorithms developed within predecessor projects AutoArch and Motion Ripper [15], the ICBR project brings novel methods of motion and gait analysis for content analysis and classification. Objectives set for the semantic modelling module include identification of overall scene type as well as individual object classification and identification. The approach is to design unsupervised classification algorithms trained on the rich and structured semantic ground-truth metadata that exploits specific media representations generated by the content adaptation module and a set of MPEG-7 audio/visual descriptors.

### 4 System Overview

As depicted in Figure 1, our ICBR system consists of four major units: Media Server, Metadata Server, a set of feature extraction and content analysis algorithms and a user-end Demonstrator comprising of a Demonstrator engine and a platform independent front-end GUI.

The Media Server, as a content provider, delivers audio and/or visual media through a QuickTime interface, either as a bandwidth-adaptive HTTP/RTSP stream for presentational purposes or as a QuickTime API's direct file access. Content analysis algorithms access media through a specific QT-based API designed to optimise various ways of access to audiovisual information. Furthermore, the modules of the content analysis algorithms that focus on the semantic analysis are able to access the Metadata Server and read/write, search and retrieve both textual and binary metadata through an MPEG-7 based API. The Metadata Server is designed to fully support the MPEG-7 descriptor set and is based on an Oracle database engine.



**Fig. 1.** ICBR Architecture shows the main entities of the system and data flow between them

The Demonstrator engine merges and controls the processes involved during search, browse and retrieval triggered by the front-end GUI. The system will support various types of interaction with multimedia database: query by example, simple text based queries, combined text and audio-visual queries, browsing based on both semantic and perceptual similarities, automated searches integrated in the media production tools, etc. The following sections describe the major parts of the ICBR system and the main content analysis modules in more detail.

#### 4.1 Media Server

The Media Server is designed to support both real-time streaming to the various modules of the ICBR system and file based access. The archive comprises an Apple Xserve content server and streaming servers including 10Tb of Raid Storage and a 100Mbps fibre channel network. The multimedia content within the ICBR project is established from the 12000 hours of Granada media digitised and transcoded into following formats:

- Master Media format, either uncompressed 601 YUV or DV which can be 525 or 625 or possibly HD formats in future
- QuickTime wrapped MPEG-4 Simple Profile, 1Mbps @25 fps (95.8kb/s) with 16bit Stereo 48k audio for LAN streaming purposes  
352X264pix/keyframe@every250f.

The content comprises of various wildlife footages and has been professionally digitised from both film tapes and analogue camera sources. The crucial value for the ICBR research is that the whole media archive has been manually annotated by production professionals. Although there are no strict rules of the language used to describe the media's content, the majority of these semantic descriptions use limited vocabulary for particular description categories like frame composition, camera work, shooting conditions, location and time, etc. In addition to that, the events and objects



are described in natural language. This valuable semantic information brings the research opportunities to a completely new level, enabling the ICBR researchers to embark upon the problem of the *semantic gap*.

## 4.2 Metadata Server

The Metadata Server unit stores, handles and facilitates querying and retrieval of both technical (e.g. descriptors, timecode, format, etc.) and semantic metadata. The underlying software and database system brings:

- A high-level semantic approach to metadata cataloguing and classification, specifically its self indexing, auto classification and multiple schema handling technologies
- Merging of semantic high-level and low-level audio-visual descriptors to enable cataloguing and classification in a novel multimedia database environment.
- Media classification: a concept node structure which is designed to hold and maintain the basic knowledge schema with its controlled vocabulary (synonym and homonym control) and cross-hierarchic, associative attributes. At the heart of this classification system is a “knowledge map” of some 50,000 concepts of which nearly 13,000 have been implemented covering many fields of knowledge: natural history, current affairs, music, etc.

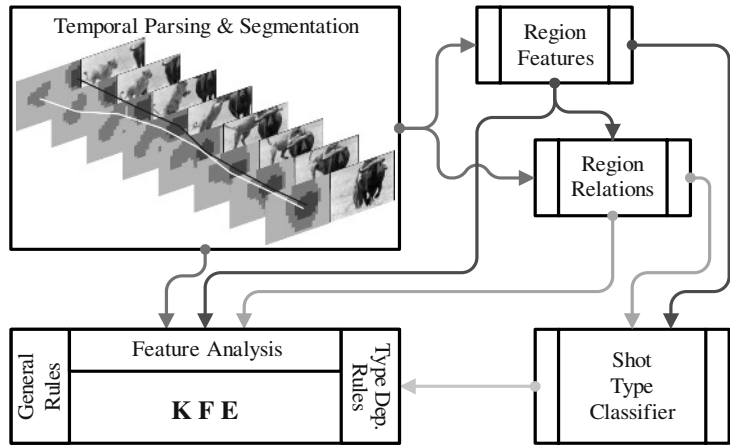
Manually generated semantic annotation is transferred from the free text format into the media classification structure, forming an irreplaceable ground truth and knowledge base.

## 4.3 Segmenter

The Segmenter module parses digitised video sequences into shots, additionally labelling camera artefacts and tape errors on the run. It utilises block-based correlation coefficients and histogram differences to measure the visual content similarity between frame pairs. In order to achieve real-time processing capability, a two-pass algorithm is applied. At first, shot boundary candidates are labelled by thresholding chi-square global colour histogram frame differences. In the second pass, more detailed analysis is applied to all candidates below a certain predetermined threshold. At this stage, hierarchical motion compensation is employed to ensure the algorithm is robust in the presence of camera and object motions. It is shown to achieve a higher recall and precision compared with the conventional shot detection techniques [16]. In addition, the algorithm detects gradual shot transitions by measuring the difference between the visual content of two distant frames. Motion estimates obtained from the shot cut detection algorithm are used to track regions of interest through the video sequence. This enables the distinction between content changes caused by gradual transitions and those caused by camera and object motions.

### 4.4 Key-Frame Extraction

In order to achieve meaningful content understanding and classification, a robust and complex, yet efficient system for video representation has to be developed. The crucial stage of that process is abstraction of a data intensive video stream into a set of still images, called key-frames that represent both the perceptual features and the semantic content of a shot in the best possible way.



**Fig. 2.** Flowchart of the rule based key-frame extraction module, where the rules applied depend upon the shot types

Current aims of the project are to extract a visual skim consisting of 8 frames to visualise the temporal content of a shot, whilst the majority of the low-level metadata will be extracted from a single key-frame.

Furthermore, information extracted in the temporal parsing module is utilised to calibrate the optimal processing load for the key-frame extractor. This information includes shot boundaries, their types and frame-to-frame difference metric, extracted using colour histogram. Due to the fact that video data demands heavy computation and that our goal is to achieve real-time processing, we need to reduce the complexity of the data input by using features extracted directly from the compressed domain video stream.

In order to tackle a subjective and adaptive intelligent key-frame extraction, we need to analyse spatio-temporal behaviour of the regions/objects present in the scene, as depicted in Figure 2. For segmentation purposes Eigen-decomposition based techniques, like nCut graph segmentation and non-linear filtering methods like anisotropic diffusion have been investigated [17]. A set of heuristic rules is to be designed in order to detect the most appropriate frame representative.

### 4.5 Motion and Gait Analysis

The animal gait analysis has focused on developing algorithms to extract quadruped gait patterns from the video footage. It is anticipated that this information will be

useful in training classifiers for low-level query by example searches as well as in semantic modelling. The algorithm generates a sparse set of points describing region trajectories in an image sequence using a KLT tracker [18]. These points are then separated into foreground and background points and the internal motion of the foreground points is analysed to extract animal's periodic motion signatures.



**Fig. 3.** Animal gait and quadruped structure from a sparse set of tracked points

A more robust tracker will allow the segmentation of the points' trajectories into many different models. This will facilitate the identification of numerous foreground objects and the separation of individual objects' motion into multiple models.

#### 4.6 Semantic Modelling

The key resource underpinning this research is a very large database of video and audio footage that will hold a variety of data types, from text to long image and audio sequences. Our aim is to automate the process of ground truth extraction by learning to recognize object class models from unlabeled and unsegmented cluttered scenes. We will build on previous works such as [19]. In addition, identification of the overall scene type is essential because it can help in identifying and modelling specific objects/animals present in the scene. Thus, a set of probabilistic techniques for scene analysis that apply Naïve Bayesian Classifiers to exploit information extracted from images will construct user-defined templates suitable for wildlife archives. Therefore the query database becomes a kind of visual thesaurus, linking each semantic concept to the range of primitive image features most likely to retrieve relevant items. We also explore techniques which allow the system to learn associations between semantic concepts and primitive features from user feedback by annotating selected regions of an image and applying semantic labels to areas with similar characteristics. As a result, the system will be capable of improving its performance with further user feedback.

#### 4.7 Audio

A primary aim on the audio side is to provide content production with 'query-by-example' retrieval of similar sounds from the Audio Database, consisting at present of

about 200Gbytes of sound. The major requirement is not to classify sounds *per se*, but to be able to retrieve sounds similar to a query sound from the database during the creative post-processing sessions when composing and laying down audio tracks on completed (silent) documentaries. Our approach, if the research proves successful, will point towards the provision of an active database. Given an initial, hopefully small, set of models trained from hand selected sounds, our thesis is that clusters of new sounds with perceptual similarities can be identified from them. Such clusters can then be used to train new models. This way we can bootstrap the Audio Database into an evolving (Active) Database. The sound clusters of new models trained “overnight” being presented to the user of the database for rejection as a non-perceptual clustering or acceptance as a new cluster with certain perceptual “features”. Provision of a textual description of the perceptual features of each new cluster would enable us to build up a complementary semantic model space. We will then be able to support both query by example sound and query by semantic (textual) description of a sound.

An automatic approach to segmentation appears essential. Our basic unit of audio sound will be a segment within a track with the content provider storing tracks as separate files. Each track in the Audio Database, mostly originally identified with a single semantic label on the CD from which it had been extracted, is likely for the sound production purposes to contain different segments of sounds related to similar sound segments across track rather than within track. These must if possible be identified, perceptually clustered and classified. We have gained invaluable experience from using the Matlab implementations of MPEG7 audio reference software developed by Michael Casey and Matthew Brand - centred upon spectrogram feature extraction, redundancy reduction and hidden Markov model classification - and will further benefit from making quantitative comparisons between our and the MPEG7 approach.

## 5 Conclusions

This paper describes the major objectives of the ICBR project and the approach taken to utilise the unique opportunity in having both large real-world multimedia database and manually annotated semantic description of media organised in a structured taxonomy. The final demonstrator will be integrated into a media production system showing functionalities of a third generation content-based multimedia retrieval system such as semantic retrieval and browsing, automatic content generation and adaptation, etc. bringing the novel multimedia management concept into a real world of media production.

**Acknowledgements.** The work reported in this paper has formed part of the ICBR project within the 3C Research programme of convergent technology research for digital media processing and communications whose funding and support is gratefully acknowledged. For more information please visit [www.3cresearch.co.uk](http://www.3cresearch.co.uk). Many thanks to David Gibson for the Figure 3.

## References

- [1] J. Calic, "Highly Efficient Low-Level Feature Extraction for Video Representation and Retrieval", PhD Thesis, Queen Mary, University of London, December 2003
- [2] M. Flickner et al., "Query by image and video content: The QBIC system", IEEE Computer 28, pp 23-32, September 1995.
- [3] A. Pentland, R.W. Picard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases", Intern. J. Comput. Vision, 18(3), pp 233-254, 1996.
- [4] A. B. Benitez, J. R. Smith, "New Frontiers for Intelligent Content-Based Retrieval", in Proc. SPIE 2001, Storage and Retrieval for Media Databases, San Jose CA, January 2001.
- [5] H. J. Zhang, A. Kankanhalli, and S. Smoliar, "Automatic Partitioning of Full-Motion Video", Multimedia Systems, Vol. 1, No. 1, pp.10-28, 1993.
- [6] B.-L. Yeo and B. Liu, "Rapid Scene Analysis on Compressed Video", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 6, pp. 533-544, December 1995.
- [7] J. Calic, E. Izquierdo, "Efficient Key-Frame Extraction and Video Analysis", 2002 International Symposium on Information Technology (ITCC 2002), 8-10 April 2002, Las Vegas, NV, USA. IEEE Computer Society, pp. 28-33, 2002,
- [8] A. Girgensohn, J. S. Boreczky, "Time-Constrained Keyframe Selection Technique", Multimedia Tools Appl., 11(3): 347-358 (2000)
- [9] M. Davis, "Knowledge Representation for Video", Proc. Of 12th National Conference on Artificial Intelligence (AAAI-94), Seattle, USA, AAAI Press, pp. 120-127, 1994.
- [10] C. Dorai, S. Venkatesh: Computational Media Aesthetics: Finding Meaning Beautiful. IEEE MultiMedia 8(4): 10-12 (2001)
- [11] M. Naphade, T. Kristjansson, B. Frey, T. S. Huang, "Probabilistic Multimedia Objects Multijets: A novel Approach to Indexing and Retrieval in Multimedia Systems", Proc. IEEE ICIP, Volume 3, pages 536-540, Oct 1998, Chicago, IL
- [12] SCHEMA Network of Excellence in Content-Based Semantic Scene Analysis and Information Retrieval [www.schema-ist.org/SCHEMA/](http://www.schema-ist.org/SCHEMA/)
- [13] COST 211quat, Redundancy Reduction Techniques and Content Analysis for Multimedia Services, [www.iva.cs.tut.fi/COST211/](http://www.iva.cs.tut.fi/COST211/)
- [14] Informedia Digital Video Library, [www.informedia.cs.cmu.edu/](http://www.informedia.cs.cmu.edu/)
- [15] Motion Ripper, [http://www.3cresearch.co.uk/3cprojects/motion\\_ripper](http://www.3cresearch.co.uk/3cprojects/motion_ripper)
- [16] Sarah V. Porter, Majid Mirmehdi, Barry T. Thomas, "Temporal video segmentation and classification of edit effects", Image Vision Comput. 21(13-14): 1097-1106, 2003.
- [17] J. Calic, B. T. Thomas, "Spatial Analysis in Key-frame Extraction Using Video Segmentation", WIAMIS'2004, April 2004, Instituto Superior Técnico, Lisboa, Portugal
- [18] D. Gibson, N. Campbell, B. Thomas, "Quadruped Gait Analysis Using Sparse Motion Information", Proc. of ICIP, IEEE Computer Society, September 2003.
- [19] M. Weber, M. Welling and P. Perona, "Unsupervised learning of models for recognition", Proc. ECCV2000, pp. 18-32, Lecture Notes in Computer Science, Springer, 2000

# Contribution of NLP to the Content Indexing of Multimedia Documents

Thierry Declerck<sup>1</sup>, Jan Kuper, Horacio Saggion, Anna Samiotou,  
Peter Wittenburg, and Jesus Contreras

Saarland University and DFKI GmbH, Stuhlsatzenhausweg 3, D66123 Saarbruecken,  
Germany,  
[declerck@dfki.de](mailto:declerck@dfki.de),  
<http://www.dfki.de/declerck>

**Abstract.** This paper describes the role *natural language processing* (NLP) can play for multimedia applications. As an example of such an application, we present an approach dealing with the conceptual indexing of soccer videos which the help of structured information automatically extracted by NLP tools from multiple sources of information relating to video content, consisting in a rich range of textual and transcribed sources covering soccer games. This work has been investigated and developed in the EU funded project MUMIS. As a second example of such an application, we describe briefly ongoing work in the context of the ESPERONTO project dealing with upgrading the actual web towards the Semantic Web (SW), including the automatic semantic indexing of web pages containing a combination of text and images.

## 1 Introduction

This paper describes the role *natural language processing* (NLP) can play in the conceptual indexing of multimedia documents which can then be searched by semantic categories instead of key words. This topic was a key issue in the MUMIS project<sup>1</sup>. A novelty of the approach developed in MUMIS is to exploit multiple sources of information relating to video content (for example the rich range of textual and transcribed sources covering soccer games). Some of the investigation work started in MUMIS is being currently pursued with the ESPERONTO project<sup>2</sup>, looking at images in web pages, and trying to apply content information to the pictures on the base of the semantic analysis of the surrounding text.

In the first part of the paper (section 2) we will propose a general discussion on the role that can be played by NLP for multimedia application. This overview is widely based on [2] and [3]. In the largest part of the paper we will exemplify

---

<sup>1</sup> MUMIS was a project within the Information Society Program (IST) of the European Union, section Human Language Technology (HLT). See for more information <http://parlevink.cs.utwente.nl/projects/mumis/>.

<sup>2</sup> ESPERONTO is a project within the 5th framework within the Information Society Program (IST) of the European Union. See for more details [www.esperonto.net](http://www.esperonto.net)

the general topic with the presentation of the MUMIS project that is concerned with the topic of multimedia indexing and searching. The presentation of this project is an extension and update of [7]. The last part of the paper will briefly sketch the ongoing work in ESPERONTO, aiming at attaching content information in images contained in web pages, by using semantic features automatically attached by NLP tools to the surrounding texts.

## 2 The Role of NLP for Multimedia Applications

[2] and [3] give an overview of the role that can be played by NLP in multimodal and multimedia systems. We summarize here the central points of those studies.

### 2.1 Multimodal and Multimedia Systems

The terms *multimedia* and *multimodal* are often source of confusion. [2] adopts the definitions as proposed in [13], which establishes a distinction between the terms *medium*, *mode* and *code*. The term *mode* (or *modality*) refers to the type of perception concerned, being for example visual, auditory or olfactory perception. The term *medium* refers to the carrier of (CD-ROM, paper etc.), to the devices (microphone, screen, loudspeakers etc.), as well to the distinct types of information (texts, audio or video sequences). The term *code* refers to the particular means of encoding information (sign languages or pictorial languages). One can speak of a multimedia system if this allows to *generate* and/or to *analyze* multimedia/multimodal information or provide some *access* to archives of multiple media. In existing applications, often the process of analysis applies only to multimodal data, whereas generation is concerned with the production of multimedia information.

### 2.2 Integration of Modalities

We speak in the case of analysis of a process of *integration of modalities*, since all the available modalities need to be merged at a more abstract level in order to take the maximal advantage of every modality involved in the application. Certain representation formalisms, as they have been defined in the context of advanced NLP (see for example [12]), can play a central role in this process of fusion. So the well-defined technique of *unification* of typed feature structures, combined with a chart parser, is used in a system described in [10]. Using this formalism allows to build a semantic representation that is common to all modalities involved in the application, unifying all the particular semantic contributions on the base of their representation in typed feature structures

### 2.3 Media Coordination

In the case of the *generation* of multimedia material including natural language, for the purpose of *multimedia presentation*, one can speak of a process of *media coordination*: it is not enough to merge various media in order to obtain a

coherent presentation of the distinct media involved. The information contained in the various media has to be very carefully put into relation if one wants to obtain real complementarities of media in the final presentation of the global information. And since systems for *natural language generation* have been always confronted with this problem of selecting and organizing various contributions for the generation of an utterance, they can provide for a very valuable model for the coordination of media in the context of the generation of multimedia presentations. A lot of systems for natural language generation are therefore said to be *plan-based*.

## 2.4 Natural Language Access to Multimedia

Multimedia repositories of moving images, texts, and speech are becoming increasingly available. This together with the needs for 'video-on-demand' systems require fine-grain indexing and retrieval mechanisms allowing users access to specific segments of the repositories containing specific types of information.

It turns out that natural language can play a multiple role. It is first easier to access information contained in the multimedia archive using queries addressed to (transcript of) audio sequences or to the subtitles (if available) associated to the videos as to analyze the pictures themselves. It is further more appealing to access visual data by means of natural language, since the latter supports more flexible and efficient queries as the query based on image features. And ultimately natural language offers a good means for condensing visual information. The selected list of projects concerned with video indexing we give below is stressing this fact: at some point always some language data will be considered to support retrieval of images or videos.

In order to support this kind of natural language access, video material was usually manually annotated with 'metadata' such as people involved in the production of the visual record, places, dates, and keywords that capture the essential content of what is depicted. Still, there are a few problems with human annotation. First, the cost and time involved in the production of "surrogates" of the programme is extremely high; second, humans are rather subjective when assigning descriptions to visual records; and third, the level of annotation required to satisfy user's need can hardly be achieved with the use of mere keywords.

## 2.5 NLP Techniques for Indexing Multimedia Material

Many research projects have explored the use of parallel linguistic descriptions of the images (either still or moving) for automatic tasks such as indexing [42], classifying [43], or understanding [44] of visual records, instead of using only content-based (or visually-based) methods in use [40]. This is partly also due to the fact that NLP technologies are more mature for extracting meaning as the technologies in use in the field of image. Content-based indexing and retrieval of visual records is based on features such as color, texture, and shape. Yet visual understanding is not well advanced and is very difficult even in closed domains. For example, visual analysis of the video of a football match can lead to the



identification of interesting “content” like a shooting scene (i.e., the ball moving towards the goal) [45], but this image analysis approach will hardly ever detect who is the main actor involved in that scene (i.e., the shooter). For accessing visual information with the help of natural language, certain systems make use of a shallow analysis of linguistic data associated with pictures, like the transcripts of audio comments or subtitles. In most of the case this is already enough in order to provide for a first classification and indexing of the visual data (see for example [11], [32] or [33]). Other systems use more sophisticated linguistic analysis, like *information extraction* (IE): the detection of *named entities* and of standard linguistic patterns can help the multimedia retrieval systems to filter out non-relevant sequences (for example the introduction of speakers in news broadcasting). An example of such systems is given in the “Broadcast News Navigator” developed at MITRE (see [15]) . The MUMIS project, described in details below, is going even further, since full IE systems are analyzing a set of so-called “collateral” (parallel) documents and produce unified conceptual annotations, including metadata information that is used for indexing the video material and supports thus concept-based queries on a multimedia archive. A similar approach is described in [23], where the domain of application is classical dance. In this work only a small set of textual documents is considered, in a monolingual setting.

### 3 MUMIS: A Multimedia Indexing and Searching Environment

MUMIS has been proposing an integrated solution to the NLP-based multimedia content indexing and search. The solution consists of using information extracted from different sources (structured, semi-structured, free, etc.), modalities (text, speech), and languages (English, German, Dutch) all describing the same event to carry out data-base population, indexing, and search. MUMIS makes an intensive use of linguistic and semantic based annotations, coupled with domain-specific information, in order to generate formal annotations of events that can serve as index for videos querying. MUMIS applies IE technologies on multilingual and multimedia information from multiple sources.

The novelty of the project was not only the use of these ‘heterogeneous’ sources of information but also the combination or cross-source fusion of the information obtained from each source. Single-document, single-language information extraction is carried out by independent systems that share a semantic model and multi-lingual lexicon of the domain. The result of all information extraction systems is merged by a process of alignment and rule-based reasoning that also uses the semantic model.

For this purpose the project makes use of data from different media (textual documents, radio and television broadcasts) in different languages (Dutch, English and German) to build a specialized set of lexicons and an ontology for the selected domain (soccer). It also digitizes non-text data and applies speech recognition techniques to extract text for the purpose of annotation. Audio material has been analyzed by Phicos [46], an HMM-based recognition system, in order

to obtain transcriptions of the football commentaries (spontaneous speech). It uses acoustic models, word-based language models (unigram and bigram) and a lexicon. For Dutch, English, and German different recognition systems have been developed. i.e. different phone sets, lexicons, and language models are used. Transcriptions for 14 German, 3 Dutch, and 8 English matches have been produced. [25] gives more details on the *automatic speech recognition* (ASR) and the transcription work done in the context of MUMIS.

The core linguistic processing for the annotation of the multimedia material consists of advanced information extraction techniques for identifying, collecting and normalizing significant text elements (such as the names of players in a team, goals scored, time points or sequences etc.) which are critical for the appropriate annotation of the multimedia material in the case of soccer. One system per language has been used or developed.

Each system delivers an XML output, an example being shown in figure 1 which serves as the input of a *merging component*, whose necessity in the project is due to the fact that MUMIS is accessing and processing multiple sources from distinct media in distinct languages. The merging tool is combining the semantically related annotations generated from those different data sources, and detect inconsistencies and/or redundancies within the combined annotations. The merged annotations are then stored in a database, where they will be combined with relevant metadata that are also automatically extracted from the textual documents.

Those annotations are delivered to the process of indexing key frames from the video stream. Key frames extraction from MPEG movies around a set of predefined time marks - result of the information extraction component - is being carried out to populate the database. JPEG key frames images are extracted that serve for quick inspection in the user interface.

Within the MUMIS user interface, the user first interacts with a web-portal to start a query session. An applet is being down-line loaded, which mainly offers a query interface. The user then enters a query that either refers to metadata, formal annotations, or both. The on-line system searches for all formal annotations that meet the criteria of the query. In doing so it will find the appropriate meta-information and/or moments in some media recording. In case of meta-information it simply offers the information in scrollable text widgets. This is done in a structured way such that different type of information can easily be detected by the user. In the case that scenes of games are the result of queries about formal annotations the user interface first presents selected video key frames as thumbnails with a direct indication of the corresponding metadata. The user can then ask for more metadata about the corresponding game or for more media data. A snapshot of the demonstrator is shown in figure 2 above.

## 4 Multimedia Indexing in the Esperanto Project

Within the Esperanto project, NLP-based annotation strategies, combining with ontologies and other knowledge bases, are applied in order to upgrade the actual Web towards the emerging Semantic Web. In this project, experiments are

7. Ein Freistoss von Christian Ziege aus 25 Metern geht ueber das Tor.  
(7. A 25-meter free-kick by Christian Ziege goes over the goal.)

```

<EVENTS>
  <TYPE>Free-kick</TYPE>
  <DISTANCE>Meter-from_(25)</DISTANCE>
  <1_PLAYER>Ziege</1_PLAYER>
  <CLASS>goal_scene_fail</CLASS>
  <ARTEFACT>Goal</ARTEFACT>
  <TIME>7:00</TIME>
</EVENTS>

<META> DOM_NAME="SOCCER" </META>

<PLAYER>
  <PLAYER_NAME>Ziege</PLAYER_NAME>
  <PLAYER_NOTE>#(3,5)</PLAYER_NOTE>
  <PLAYER_POS>#4 ##3</PLAYER_POS>
  <PLAYER_NUMBER>##17</PLAYER_NUMBER>
  <PLAYER_AGE>##28</PLAYER_AGE>
  <PLAYER_CLUB>##FC Middlesbrough</PLAYER_CLUB>
  <PLAYER_NUMB_PLAYS>##52</PLAYER_NUMB_PLAYS>
</PLAYER>

```

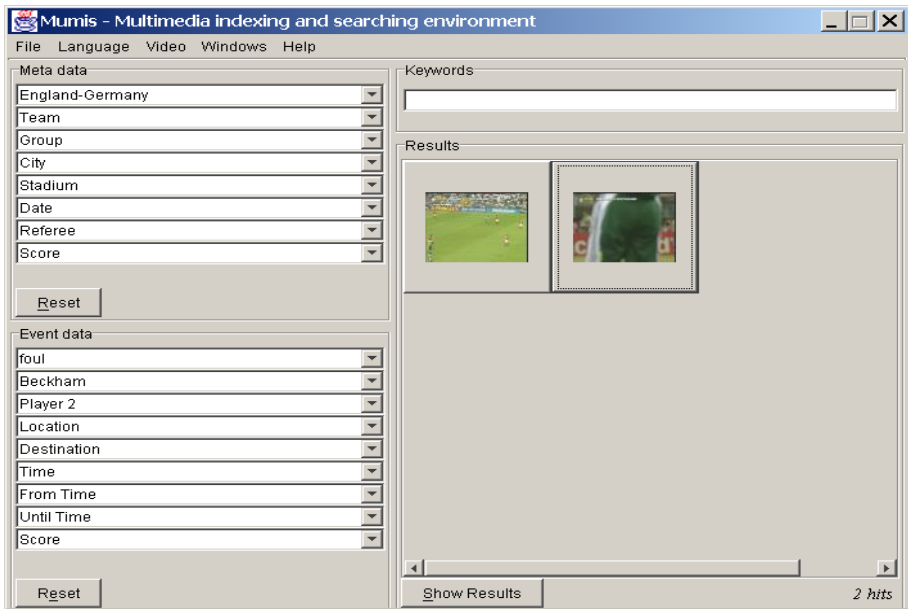
**Fig. 1.** Example of the XML encoding of the result of the automatic extraction from a sentence of a relevant event, with its associated relations entities and relations. The information about the player is dynamically included from the processing of 2 structured texts reporting on the same game, marked with # and ## respectively.

also conducted on the use of Semantic Web annotation structures, eventually consisting of complex ontology-based frames (or templates), that are associated to parts of text surrounding pictures to the indexing of the pictures themselves. Work is still not advanced enough in order to be reported in detail this paper, but actual results show that the main problem will consist in automatically detecting the parts of text related to the pictures. Here the caption of the picture, as well as the name of the image in the html document can offer a support for filtering the relevant semantic annotation from the surrounding texts.

## 5 Conclusions

The MUMIS experience has shown that NLP can contribute in defining semantic structures of multimedia contents, at the level proposed by domain-specific IE analysis. The full machinery of IE, combined with ASR (and in the future with Image Analysis, so for example with the actual results of the Schema Reference platform<sup>3</sup> can be used for multimedia contents development and so efficiently

<sup>3</sup> See <http://www.schema-ist.org/SCHEMA/> and also the contribution on the SCHEMA reference platform in this volume.



**Fig. 2.** MUMIS User Interface. Thumbnails proposed for the query “Show the fouls comited by Beckham in the game England-Germany”

support cross-media (and cross-lingual) information retrieval and effective navigation within multimedia information interfaces, thus simplifying the access to content (and knowledge) distributed over multiple documents and media, which are also increasingly available on the Web. The ESPERONTO project is currently porting some of the experiences gained in former projects on content indexing of multimedia documents in the Semantic Web framework. We will try to integrate part of the work described above within the SCHEMA reference platform.

Work that remains to do consist in fully integrating results of image/video content analysis with the semantic analysis of text/transcripts, towards a full Semantic Web annotation services for multimedia documents.

**Acknowledgements.** This research has in part been supported by EC grants IST-1999-10651 for the MUMIS project and IST-2001-34373 for the ESPERONTO project.

## References

1. Adani N., Bugatti A., Leonardi R., and Migliorati P. Semantic description of multimedia documents: the Mpeg-7 approach. In *Proceedings of the Conference on Content-Based Multimedia Indexing, CBMI-2001, Brescia, 2001*.
2. André E. Natural Language in Multimedia/Multimodal Systems. In Mitkov R. (ed.), *Handbook of Computational Linguistics*, Oxford, 2000.

3. André E. The Generation of Multimedia Presentations. In Handbook of Natural Language Processing, Marcel Dekker, 2000.
4. Assfalg J., Bertini M., Colombo C., and Del Bimbo A. Semantic annotations of sports videos. In Proceedings of the Conference on Content-Based Multimedia Indexing, CBMI-2001, Brescia, 2001
5. Cunningham H. Information Extraction: A user Guide, Research Report CS-99-07, Department of Computer Science, University of Sheffield, May 1999.
6. Day N. MPEG-7 Applications: Multimedia Search and Retrieval. In Proceedings of the First International Workshop on Multimedia Annotation, MMA-2001, 2001.
7. Declerck T., Wittenburg P., Cunningham H. The Automatic Generation of Formal Annotations in a Multimedia Indexing and Searching Environment. Proceedings of the Workshop on Human Language Technology and Knowledge Management, ACL-2001, 2001.
8. Declerck T. A set of tools for integrating linguistic and non-linguistic information. Proceedings of SAAKM 2002, ECAI 2002, Lyon.
9. Djoerd H., de Jong F., Netter K. (Eds). 14th Twente Workshop on Language Technology, Language Technology in Multimedia Information Retrieval, TWLT 14, Enschede, Universiteit Twente, 1998.
10. Johnston M. Unification-based Multimodal Parsing, In Proceedings of the 17th International Conference on Computational Linguistics, COLING-98, 1998.
11. de Jong F., Gauvin J., Hiemstra D., Netter K. Language-Based Multimedia Information Retrieval. In Proceedings of the 6th Conference on Recherche d'Information Assistée par Ordinateur, RIAO-2000, 2000. Indexing Workshop (CBMI2001), 2001.
12. Krieger H.-U., Schaefer U. TDL – a type description language for constraint-based grammars. In Proceedings of the 15th International Conference on Computational Linguistics, COLING-94, 1994.
13. Maybury M. Multimedia Interaction for the New Millenium. In Proceedings of Eurospeech 99, 1999.
14. McKeown K. Text generation, Cambridge University Press, 1985.
15. Merlino A., Morey D., Maybury M. Broadcast News Navigation using Story Segments. ACM International Multimedia Conference, 1997.
16. Miller, G.A. WordNet: A Lexical Database for English. Communications of the ACM 11. 1995.
17. Moore J., Paris C. Planning Text for Advisory Dialogues. In Proceedings of the 27th ACL, Vancouver, 1989.
18. Sixth Message Understanding Conference (MUC-6), Morgan Kaufmann, 1995.
19. Seventh Message Understanding Conference (MUC-7),  
<http://www.muc.saic.com/>, SAIC Information Extraction, 1998.
20. Naphade Milid R. and T.S. Huang. Recognizing high-level concepts for video indexing. In Proceedings of the Conference on Content-Based Multimedia Indexing, CBMI-2001, Brescia, 2001. Extraction and Navigation System. In Proceedings of the 6th Conference on Recherche d'Information Assistée par Ordinateur, RIAO-2000, 2000.
21. Saggion H. , Cunningham H., Bontcheva K., Maynard D, Ursu C. Hamza O. and Wilks Y. Access to Multimedia Information through Multisource and Multilanguage Information Extraction. 7th Workshop on Applications of Natural Language to Information Systems (NLDB 2002), 2002.
22. Salembier P. An overview of Mpeg-7 multimedia description schemes and of future visual information challenges for content-based indexing. In Proceedings of the Conference on Content-Based Multimedia Indexing, CBMI-2001, Brescia, 2001.

23. Salway A., Talking Pictures: Indexing and Representing Video with Collateral Texts. In Hiemstra D., de Jong F., Netter K. (Eds), *Language Technology in Multimedia Information Retrieval* (Proceedings of the 14th Twente Workshop on Language Technology, TWLT 14), Enschede, Universiteit Twente, 1998.
24. Staab S., Maedche A., Handschuh S. An Annotation Framework for the Semantic Web. In *The First International Workshop on Multimedia Annotation*, Tokyo, Japan, 2001.
25. Wester M., Kessens J.M. and Strik H. Goal-directed ASR in a Multimedia Indexing and Searching Environment (MUMIS). *Proceedings of the 7th International Conference on Spoken Language Processing (ICLSP2002)*, 2002.
26. EUR: <http://www.foyer.de/euromedia/>
27. GDA: <http://www.csl.sony.co.jp/person/nagao/gda/>
28. INF: <http://www.informedia.cs.cmu.edu/>
29. ISI: <http://www.wins.uva.nl/research/isis/isisNS.html>
30. ISLE: [http://www.ilc.pi.cnr.it/EAGLES/ISLE\\_Home\\_Page.htm](http://www.ilc.pi.cnr.it/EAGLES/ISLE_Home_Page.htm)
31. NSF: <http://www.nsf.gov/od/lpa/news/press/pr9714.htm>
32. OLI: <http://twentyone.tpd.tno.nl/olive>
33. POP: <http://twentyone.tpd.tno.nl/popeye>
34. SUR: <http://www-rocq.inria.fr/nastar/MM98/node1.html>
35. THI: <http://www.dcs.shef.ac.uk/research/groups/spandh/projects/thisl>
36. UMA: <http://ciir.cs.umass.edu/research/>
37. UNL: [http://www.ias.unu.edu/research\\_prog/science\\_technology/universalnetwork\\_language.html](http://www.ias.unu.edu/research_prog/science_technology/universalnetwork_language.html)
38. VIR: <http://www.virage.com/>
39. COL: <http://www.cs.columbia.edu/hjing/sumDemo>
40. Veltkamp R. and Tanase M. Content-based Image Retrieval Systems: a survey. Technical report UU-CS-2000-34, Utrecht University, 2000.
41. Chang S.F., Chen, W., Meng H.J., Sundaram H. and Zhong D. A Fully Automated Content-based Video Search Engine Supporting Spatio Temporal Queries. *IEEE Transactions on Circuits and Systems for Video Technology*, 1998.
42. Netter K. Pop-Eye and OLIVE. Human Language as the Medium for Cross-lingual Multimedia Information Retrieval. Technical report, Language Technology Lab. DFKI GmbH, 1998.
43. Sable C. and Hatzivassiloglou V. Text-based approaches for the categorization of images. *Proceedings of ECDL*, 1999.
44. Srihari R.K. Automatic Indexing and Content-Based Retrieval of Captioned Images, *Computer* 28/9, 1995.
45. Gong Y., Sin L.T., Chuan C.H., Zhang H. and Sakauchi M. Automatic Parsing of TV Soccer Programs. *Proceedings of the International Conference on Multimedia Computing and Systems (IEEE)*, 1995.
46. Steinbiss V., Ney H., Haeb-Umbach R., Tran B.-H., Essen U., Kneser R., Oerder M., Meier H.-G., Aubert X., Dugast C. and Geller D. The Philips Research System for Large-Vocabulary Continuous-Speech Recognition. In *Proc. of Eurospeech '93*, 1993.

# The CIMWOS Multimedia Indexing System

Harris Papageorgiou and Athanassios Protopapas

Institute for Language and Speech Processing  
Artemidos 6 and Epidavrou, Athens 15125, Greece  
{xaris,protopap}@ilsp.gr

**Abstract.** We describe a multimedia, multilingual and multimodal research system (CIMWOS) supporting content-based indexing, archiving, retrieval and on-demand delivery of audiovisual content. CIMWOS (Combined IMage and Word Spotting) incorporates an extensive set of multimedia technologies by seamless integration of three major components – speech, text and image processing – producing a rich collection of XML metadata annotations following the MPEG-7 standard. These XML annotations are further merged and loaded into the CIMWOS Multimedia Database. Additionally, they can be dynamically transformed for interchanging semantic-based information into RDF documents via XSL stylesheets. The CIMWOS Retrieval Engine is based on a weighted boolean model with intelligent indexing components. A user-friendly web-based interface allows users to efficiently retrieve video segments by a combination of media description, content metadata and natural language text. The database includes sports, broadcast news and documentaries in three languages.

## 1 Introduction

The advent of multimedia databases and the popularity of digital video as an archival medium pose many technical challenges and have profound implications for the underlying model of information access. Digital media assets are proliferating and most organizations, large broadcasters as well as SMEs are building networks and technology to exploit them. Traditional broadcasters, publishers and Internet content providers are migrating into increasingly similar roles as multimedia content providers. Digital technology today allows the user to manipulate or interact with content in ways not possible in the past. The combination of PCs and networks allows the individual to create, edit, transmit, share, aggregate, personalize and interact with multimedia content in increasingly flexible ways. The same technology allows content to be carried across different platforms. In fact, much of the information that reaches the user nowadays is in digital form: digital radio, music CDs, MP3 files, digital satellite and digital terrestrial TV, personal digital pictures and videos and, last but not least, digital information accessed through the Web. This information is heterogeneous, multimedia and, increasingly, multi-lingual in nature.

The development of methods and tools for content-based organization and filtering of this large amount of multimedia information reaching the user through many and different channels is a key issue for its effective consumption and enjoyment. There are several projects aiming at developing advanced technologies and systems to tackle the problems encountered in multimedia archiving and indexing [1], [2], [3].

The approach taken in CIMWOS was to design, develop and test an extensive set of multimedia technologies by seamless integration of three major components – speech, text and image processing – producing a rich collection of XML metadata annotations and allowing the user to store, categorize and retrieve multimedia and multi-lingual digital content across different sources (TV, radio, music, Web).

This paper is organized as follows: in the next three sections we focus on technologies specific to Speech, Text, and Image respectively. These technologies incorporate efficient algorithms for processing and analyzing relevant portions from various digital media and thus generating high-level semantic descriptors in the metadata space. CIMWOS architecture for the integration of all results of processing is presented in the following section. Evaluation results are reported in the next section and finally future work is drawn in the last section.

## 2 Speech Processing Component

When transcribing broadcast news we are facing clean speech, telephone speech, conference speech, music, and speech corrupted by music or noise. Transcribing the audio, i.e. producing a (raw) transcript of what is being said, determining who is speaking, what topic a segment is about, or which organizations are mentioned, are all challenging problems due to the continuous nature of the data stream. One speaker may also appear many times in the data.

We would also like to determine likely boundaries of speaker turns based on non-speech portions (silence, music, background-noise) in the signal, so that regions of different nature can be handled appropriately.

The audio stream usually contains segments of different acoustic and linguistic nature exhibiting a variety of difficult acoustic conditions, such as spontaneous speech (as opposed to read or planned speech), limited bandwidth (e.g. telephone interviews), speech in presence of noise, music or background speakers. Such adverse background conditions lead to significant degradation in performance of the speech recognition systems if appropriate countermeasures are not taken. The segmentation of the continuous stream into homogeneous sections (speaker and/or acoustic/background conditions) also poses serious problems. Successful segmentation however, forms the basis for further adaptation and processing steps. Consequently, adaptation to the varied acoustic properties of the signal or to a particular speaker, and enhancements of the segmentation process, are generally acknowledged to be key areas for research to render indexing systems usable for actual deployment. This is reflected by the effort and the number of projects dedicated to advance the state-of-the-art in these areas.

In CIMWOS, all of these tasks are being taken care by the Speech Processing Component (SPC). The SPC comprises the Speaker Change Detection module (SCD), Automatic Speech Recognition engine (ASR), Speaker Identification (SID) and Speaker Clustering (SC). The ASR engine is a real-time, large vocabulary, speaker-independent, gender-independent, continuous speech recognizer [4], trained in a wide variety of noise conditions encountered in the broadcast news domain.



### 3 Text Processing Component

After processing the audio input, text-processing tools operate on the textual stream produced by the SPC and perform the following tasks: Named Entity Detection (NED), Term Extraction (TE), Story Detection (SD) and Topic Classification (TC).

#### 3.1 Named Entity Detection (NED) and Term Extraction (TE)

The task of the Named Entity Detection (NED) module is to identify all named locations, persons and organizations in the transcriptions produced by the ASR component. CIMWOS uses a series of basic language technology building blocks, which are modular and combined in a pipeline [5]. An initial finite state preprocessor performs tokenization on the output of the speech recognizer. A part-of-speech tagger trained on a manually annotated corpus and a lemmatizer carry out morphological analysis and lemmatization. A lookup module matches name lists and trigger-words against the text, and, eventually, a finite state parser recognizes NEs on the basis of a pattern grammar. Training the NED module includes populating the gazetteer lists and semi-automatically extracting the pattern rules. A corpus of 100.000 words of gold transcriptions of broadcast news per language (English, Greek) already tagged with the NE classes was used to guide system training and development.

The term extraction (TE) module involves the identification of single or multi-word indicative keywords (index terms) in the output of the ASR. Systems for automatic term extraction using both linguistic and statistical modeling are reported in the literature [6]. The term extractor in CIMWOS follows the same architecture: linguistic processing is performed through an augmented term grammar, the results of which are statistically filtered using frequency-based scores.

NED and TE modules were tested on pre-selected sequences of Greek broadcasts [10]. The NED module obtained a 79-80% precision and a 52-53% recall for locations and persons. Lower recall is due to missing proper names in the vocabulary of the ASR engine as also the diversity of domains found in broadcasts. The TE module automatically annotated the same data, scoring a 60% recall and a 35% precision.

#### 3.2 Story Detection (SD) and Topic Classification (TC)

The basis of the Story Detection (SD) and Topic Classification (TC) modules is a generative *mixture-based Hidden Markov Model* (HMM). The HMM includes one state per topic and one state modeling general language, that is, words not specific to any topic. Each state models a distribution of words given the particular topic. After emitting a single word, the model re-enters the beginning state and the next word is generated. At the end of the story, a final state is reached. SD is performed running the resulting models on a fixed-size sliding window, noting changes in topic-specific words as the window moves on. The result of this phase is a set of 'stable regions' in which topics change only slightly or not at all. Building on the story-boundaries thus located, TC classifies the sections of text according to the set of topic models (and a general language model). The modeled inventory of topics is a flat, Reuters-derived structure containing about a dozen of main categories and several sub-categories. The

annotators had the freedom to add one level of detail to each topic during transcription. Several iterations were needed to arrive at a level of topics shallow enough to provide reasonable amounts of training data per topic but still fine-grained enough to allow for flexibility and detail in queries.

## 4 Image Processing Component

The Image Processing Component (IPC) consists of modules responsible for video segmentation and keyframe extraction; detection and identification of faces at any location, scale, and orientation; recognition of objects given knowledge accumulated from a set of previously seen “learning views”; and video text detection and recognition. A brief description of these modules is given in the following subsections.

### 4.1 Face Detection and Identification (FD/FI)

The FD and FI modules spot faces in keyframes and associate them with names. Given a keyframe extracted from the video by the AVS module, the FD module will attempt to determine whether or not there are any faces in the image and, if present, to return the image location and extent of the face. In spite of expanding research in the field, FD remains a very challenging task because of several factors influencing the appearance of the face in the image. These include identity, pose (frontal, half-profile, profile), presence or absence of facial features such as beards, moustaches and glasses, facial expression, occlusion and imaging conditions. In face recognition and identification, intra-personal variation (in the appearance of a single individual due to different expressions, lighting, etc.) and extra-personal variation (variations in appearance between persons) are of particular interest.

In CIMWOS, both FD and FI modules [7] are based on Support Vector Machine models. The FD module was trained on an extensive database of facial images with a large variation in pose and lighting conditions. Additionally, a semantic base of important persons was compiled for FI training. During the FI stage, faces detected by the FD module in the keyframes are compared to the model corresponding to each face in the semantic base. Scores resulting from the comparison guide the module to identify a candidate face or classify it as “unknown.”

### 4.2 Object Recognition (OR)

Object Recognition (OR) is used to spot and track pre-defined “objects” of interest. The objects can be scenes, logos, designs, or any user-selected image parts of importance, manually delineated on a set of “example views,” which are used by the system to create object models. In CIMWOS, the object’s surface is decomposed in a large number of regions (small, closed areas on the object’s surface) automatically extracted from the keyframes. The spatial and temporal relationships of these regions are acquired from several example views. These characteristics are incorporated in a model, which can thus be gradually augmented as more object examples are assimilated [8].

This methodology has two fundamental advantages: first, the regions themselves embed many small, local pieces of the object appearance at the pixel level. Thus, even in the case of occlusion or clutter, they can reliably be associated with training examples, since a subset of the object's regions will still be present. The second strong point is that the model captures the spatio-temporal order inherent in the set of individual regions and requires it to be present in the recognition view. This way the model can reliably and quickly accumulate evidence about the identity of the object in the recognition view, even in cases where only a small amount of recognized regions is found (e.g.: strong occlusion, difficult illumination conditions, which might make many individual regions hard to spot).

Thanks to the good degree of viewpoint invariance of the regions, and to the strong model and learning approach developed, the OR module can cope with 3-D objects of general shape, requiring only a limited number of learning views. This way objects can be recognized in a wide range of previously unseen viewpoints, in possibly cluttered, partially occluded, views.

### 4.3 Video Text Detection and Recognition (TDR)

Text in a video stream may appear in captions produced by the broadcaster, or in labels, logos etc. The goal of the TDR module is to efficiently detect and recognize these textual elements by integrating advanced Optical Character Recognition (OCR) technologies. Since text often conveys semantic information directly relevant to the content of the video (such as a politician's name or an event's date and location), TDR is recognized as a key component in an image/video annotation and retrieval system. However, text characters in video streams usually appear against complex backgrounds and may be of low resolution, and/or of any colour or greyscale value. The direct application of conventional OCR technology has been shown to lead to poor recognition rate. Better results are obtained if efficient location and segmentation of text characters occurs before the OCR stage.

The CIMWOS TDR module is based on a statistical framework using state-of-the-art machine learning tools and image processing methods [9]. Processing consists of four stages: Text detection aims at roughly and quickly finding blocks of image that may contain a single line of text characters. False alarms are removed during the text verification stage, on the basis of a Support Vector Machine model. Text segmentation uses a Markov Random Field model and an Expectation Maximization algorithm to extract pixels from text images belonging to characters, with the assumption that they have the same colour/grey scale value. During the final processing stage, all hypotheses produced by the segmentation algorithm are processed by the OCR engine. A string selection is made based on a confidence value, computed on the basis of character recognition reliability and a simple bigram language model.

## 5 Integration Architecture

The critical aspect of indexing the video segment is the integration of image and language processing. Each scene is characterized by the metadata that appear in it. All processing modules in the corresponding three modalities (Audio, Image and Text)

converge to a textual XML metadata annotation scheme following the MPEG-7 descriptors. These XML metadata annotations are further processed, merged and loaded into the CIMWOS Multimedia DataBase.

The merging Component of CIMWOS is responsible for the amalgamation of the XML annotations and the creation of one self-contained object that is compliant with the CIMWOS Database. Additionally, the resulting object can be dynamically transformed into RDF (<http://www.w3.org/RDF/>) documents, for interchanging semantic-based information, via XSL stylesheets (<http://www.w3.org/Style/XSL>).

The CIMWOS Integration architecture (Fig 1) follows a N-tier scenario by integrating a data services layer (storage & retrieval of metadata), a business services layer incorporating all remote multimedia processors (audio, video and text intelligent engines), and a user services layer which basically includes the user interface (UI) and web access forms.

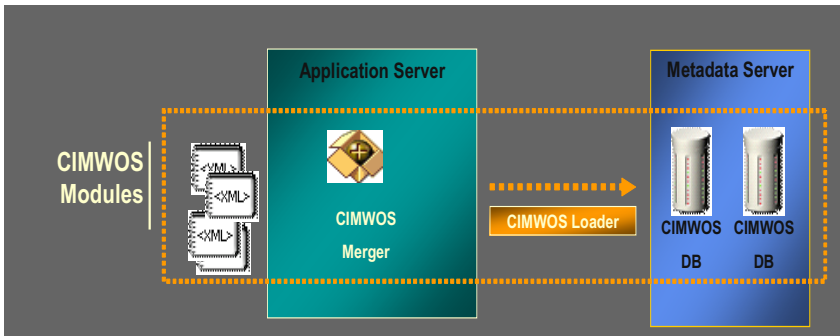


Fig. 1. CIMWOS Architecture

CIMWOS retrieval engine is based on a weighted Boolean model. Each video segment is represented by its XML metadata annotation object. Boolean logic is effectively combined with a metadata-weighting scheme for the similarity measure which is a function of three factors: metadata-level weights based on the calculated overall precision of each multimedia processor, value-level statistical confidence measures produced by the different engines and  $tf*idf$  scores for all textual elements (spoken words). A more detailed description of the retrieval engine can be found in [10].

## 6 Evaluation

In this section, we focus on the assessment of the Multimedia Information Retrieval (MIR) CIMWOS system response to user queries. The evaluation of each contributing module is reported elsewhere [10]. The accuracy of the integrated environment's retrieval capabilities were tested on Greek news broadcasts, on the basis of groundtruth annotations created by users of different expertise.

## 6.1 Evaluation Materials and Method

Evaluation and validation were divided in two phases. During the first phase, the overall success in retrieval of passages relevant to a particular topic was assessed on pre-selected sequences of Greek broadcasts. During the second phase of the validation the retrieval task was repeated, this time on new video material, for which boundaries of relevant stories for each topic had been previously identified by human annotators.

A group of three individuals (2 men, 1 woman) was set up to carry our validation from an end-user viewpoint. They had different extensive backgrounds in working with multimedia objects: an archivist of audiovisual material for the Greek state television, a postgraduate student in journalism and media studies, and a historian specialized in the creation and management of historical archives.

We used 35.5 hours of digital video during the two validation phases. The material consisted of Greek news broadcasts by state and private TV networks between March 2002 and July 2003. Each video file corresponded to one news broadcast. For the first validation phase we used 15 videos (henceforth, Collection A) of approximately 18 hours total duration, captured in BETA SP and transcribed in MPEG-2. For the second phase we used 15 broadcasts (henceforth, Collection B) amounting to approximately 17 hours of video, captured via standard PC TV cards in MPEG-2. Bibliographic data (creator, duration and format, etc.) were recorded in the project's media description format. The fact that the video collection spanned a relatively long time period ensured the diversity of the news stories presented in the broadcasts.

Gold annotations of each collection were created by the user group, using XML-aware editors for the compilation of groundtruth data. The CIMWOS query interface offers different views of the results, and each user was responsible for storing results in reusable XML files containing information about the criteria used in each query. A user familiar with the test collection was responsible for the generation of a list of interesting topics. The user then located manually all relevant sequences, allowing the association of start and end timestamps with each topic.

During search, users generated queries for each topic of the list based on their own judgment, using combinations of Terms, Named Entities and/or ASR Text. Users formed 5 queries, on average, for each topic. Using the HTML interface, they could browse the results to assess their overall satisfaction with the results.

Users found the returned passages to be short and fragmented in the results from queries on Collection A videos. This was attributed to the fact that passage identification was based on automatic segmentation and clustering by the speech recognition module, based on speaker turns. For Collection B testing, a different approach was taken to produce more intuitive passage segmentation. Manual identification of relevant segments was again undertaken, but this time segmentation was based on the *stories* contained in each broadcast. The story boundaries were then aligned to the ASR transcriptions, thus avoiding the temporally short passages observed phase A.

## 6.2 Results

We collected each version for each user's query as an XML file, and tested against gold data. Results are shown in Table 1.

Further testing included filtering out results that scored less than 60% in the CIMWOS DB ranking system. Although a decrease in the system's recall was observed in the case of Collection B, this filter significantly increased precision in both validation phases.

The MIR validation phase confirmed the hypothesis that not all metadata annotations are equally important in terms of retrieval accuracy and users' satisfaction. The experiments showed that accurate TDR and OR combined with a state-of-the-art ASR engine (10-20% WER in BNs) can adequately support most retrieval tasks specifically in case the search unit is the story and not the passage (speaker turns). Moreover, FD/FI information should be combined with ASR/NED and SID (speaker identification) results in order to increase the accuracy of the named persons.

**Table 1.** Retrieval results on Greek video collections

	Precision	Recall	F-measure
Collection A	34.75	57.75	43.39
Collection A + 60% Filter	45.78	53.52	49.35
Collection B	44.78	50.24	47.36
Collection B + 60% Filter	64.96	37.07	47.20

## 7 Future Work

The three major components – speech, text and image – of the multimodal indexing subsystem incorporated in the CIMWOS integrated platform produce a rich set of metadata indices following MPEG-7 descriptors, allowing for flexibility in retrieval tasks. Our future work focuses on semantically enriching the contents of multimedia documents with topic, entity and fact information relevant to user profiles; developing suitable cross-language cross-media representations; and building classification and summarization capabilities incorporating cross-language functionality (cross-language information retrieval, categorization and machine translation of indicative summaries) based on statistical machine translation technology.

**Acknowledgments.** This work was supported in part by shared-cost research and technological development contract IST-1999-12203 with the European Commission (project CIMWOS; see [www.xanthi.ilsp.gr/cimwos/](http://www.xanthi.ilsp.gr/cimwos/)).

CIMWOS subsystems and components developed at the Institute for Language & Speech Processing (Greece), Katholieke Universiteit Leuven (Belgium), Eidgenössische Technische Hochschule Zurich (Switzerland), Sail Labs Technology AG (Austria), and Institut Dalle Molle d'Intelligence Artificielle Perceptive (Switzerland). User requirements compiled by Canal+ Belgique (Belgium).

## References

1. Wactlar, H., Olligschlaeger, A., Hauptmann, A., Christel, M. "Complementary Video and Audio Analysis for Broadcast News Archives", Communications of the ACM, 43(2), pp. 42-47, February, 2000
2. Michael R. Lyu , Edward Yau , Sam Sze. "Video and multimedia digital libraries: A multilingual, multimodal digital video library system", In Proc. Of the 2<sup>nd</sup> ACM/IEEE-CS joint conf. On Digital Libraries, July 2002, pp.145-153.
3. Sankar A., Gadde R.R. and Weng F. "SRI's broadcast news system – Toward faster, smaller and better speech recognition", In Proc. Of the DARPA Broadcast News Workshop, 1999, pp.281-286
4. Kubala, F., Davenport, J., Jin, H., Liu, D., Leek, T., Matsoukas, S., Miller, D., Nguyen, L., Richardson, F., Schwartz, R. & Makhoul, J. (1998). The 1997 BBN BYBLOS System applied to Broadcast News Transcription. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*. Lansdowne VA.
5. Demiros, I., Boutsis, S., Giouli, V., Liakata, M., Papageorgiou, H. & Piperidis, S. (2000) Named Entity Recognition in Greek Texts. In *Proceedings of Second International Conference on Language Resources and Evaluation-LREC2000* (pp.1223-1228). Athens, Greece.
6. Jacquemin, C. & Bourigault, D. (2003). Term Extraction and Automatic Indexing. In Mitkov R., (Ed.), *Handbook of Computational Linguistics*, (pp. 599-615). Oxford University Press, Oxford.
7. Cardinaux F. & Marcel S. (2002). Face Verification Using MLP and SVM. In Neurosciences et Sciences de l'Ingenieur, France.
8. Ferrari, V., Tuytelaars, T., & Van Gool, L. (2003). Wide-baseline multiple-view correspondences. In *Proc. of IEEE Computer Vision and Pattern Recognition*. Madison, USA.
9. Odobez, J. M., & Chen, D. (2002) Robust Video Text Segmentation and Recognition with Multiple Hypotheses. In *Proc. of the International Conference on Image Processing*.
10. Papageorgiou H., Prokopidis, P., Demiros I., Hatzigeorgiou N. & G. Carayanis. (2004). CIMWOS: A Multimedia retrieval system based on combined text, speech and Image processing. In *Proceedings of the RIAO Conference (RIAO-2004)*, Avignon, France

# Image Retrieval Interfaces: A User Perspective

John P. Eakins<sup>1</sup>, Pam Briggs<sup>2</sup>, and Bryan Burford<sup>2</sup>

<sup>1</sup>School of Informatics      <sup>2</sup>Division of Psychology  
University of Northumbria, Newcastle NE1 8ST, U K  
{john.eakins, p.briggs, b.burford}@unn.ac.uk

**Abstract.** Surprisingly little is known about how different users conduct image searches. As a result, even the most sophisticated systems available have limited appeal to the end-user. This paper describes a study eliciting user requirements for future image databases through an online questionnaire. 125 experienced image searchers were questioned about the functions and modes of interaction that they currently use, and those they would like to see in future systems. The results of this survey, and their implications for retrieval systems design, are discussed in some detail.

## 1 Introduction

### 1.1 Background

Digital images are now an important resource both for work and for leisure. Traditional image databases rely on text or metadata as their primary access mode, typically using structured thesauri or classification systems such as the one described by Bjarnestam [1]. Such systems are in widespread use, but suffer from the drawback that assigning keywords is inherently both labour-intensive and subjective [2].

An alternative access mode is provided by content-based image retrieval (CBIR) techniques [3]. Most of these are geared towards retrieval by image appearance, using automatic extraction and comparison of image features such as colour, texture, shape, and spatial layout. The results of such techniques can look quite impressive - a visual query consisting of a picture of a golden sunset can retrieve a whole screenful of similar-looking golden sunsets. Unfortunately, the evidence of user studies [4] suggests that such a facility is of limited use in meeting image users' real needs.

One of the problems currently holding back the development of CBIR technology [5] is the difficulty of specifying a visual query. Most CBIR systems work on the *query-by-example* principle: a query image is input to the system, which then searches for other images similar to the query. The user can often specify the relative importance of different matching criteria, such as colour, texture and shape. However, this process suffers from a number of limitations:

- It is impossible to start a search if no suitable query image can be found.
- Using a single query image can *overspecify* the search. If, for example, a picture of a car containing two passengers is used as the query image, the passengers become part of the search whether the user wants this or not.
- Search parameters such as the relative importance of colour, shape or texture are not as intuitive as text, and users find it difficult to manipulate them to improve search results.



A number of general-purpose CBIR systems are (or have been) commercially available, including IBM's QBIC (Query By Image Content) [6] and more recently LTU Technology's ImageFinder (<http://www.ltutechnology.com>). While these systems exploit sophisticated algorithms for image analysis and similarity matching, it is not clear that they really address user needs. For example, users can use QBIC to conduct colour based searches of the art works in the Hermitage Museum (<http://www.heritagemuseum.org/fcgi-bin/db2www/qbicSearch.mac/qbic?sellLang=English>) - but it is by no means clear why anyone would wish to do so. The problem is that we know relatively little about the context of use for image retrieval: Who uses the retrieval system? What kinds of images are they interested in? What elements do they find important in an image and how will they use the images they find? Many of the latest developments in image retrieval have been technology- rather than user-driven. Yet without a real understanding of the cognitive and contextual needs of the end user, many state-of-the art systems risk failure.

This is where user-centred design principles are important. We know that users have complex requirements which are both linguistic and visual [4]. Whether or not current technology can adequately support such a range of queries is open to question. But the user requirements for such technology can be explored, so that those working on developing functionality have targets to aim for.

One approach to understanding user needs is to develop frameworks for understanding image retrieval such the taxonomy of image content proposed by Burford et al [7], and the taxonomy of image use developed by Conniss et al [8]. In our current study, these frameworks will be used to ask questions about what types of image content are important for what types of image use, and further to ask what interface elements should be made available to support the process of image query.

## 1.2 A Taxonomy of Image Content

The taxonomy of image content developed by Burford et al [7] was based on an extensive survey of the computer science, art history and psychology literatures, and subsequently validated with a small set of professional image users. It identifies ten distinct categories, as follows:

**Perceptual primitives** include the lowest levels of visual content, such as edges, texture elements and colour, including colour histograms and measures of image sharpness.

**Geometric primitives** refer to the simplest *structures* extractable from images, such as lines, curves, or recognizable geometric shapes.

**Visual relationships** refer to the spatial arrangement of objects in a scene in two dimensions; **visual extension** to the arrangement of objects in the third dimension, including the presence of depth cues.

**Semantic units** are the names of objects, or classes of object, present in a scene. These may be *general* (types of object or material, such as "horse" or "sand") or *specific* (individual entities such as "Abraham Lincoln" or "the Eiffel Tower").

**Abstraction** refers to content which is not directly present in the image, but needs to be inferred from background knowledge and experience. There are four levels of abstraction in the taxonomy:

**Contextual abstraction** refers to non-visual information derived from knowledge of the environment, such as whether an image represents a day or night scene.

*Cultural abstraction* refers to aspects of a picture that can be inferred only with the help of specific cultural knowledge, such as understanding the religious significance of an image of a procession.

*Emotional abstraction* refers to the emotional responses evoked by an image. While such responses may be universal, they are typically idiosyncratic, varying from viewer to viewer.

*Technical abstraction* refers to aspects requiring specific technical expertise to interpret, such as whether a patient's X-ray shows evidence of lung cancer.

Finally, *metadata* refers to terms which describe the image itself, such as its size, type (painting, photograph etc) or creator.

### 1.3 Uses of Image Data

In order to fully understand the requirements of the end-user, we also need a taxonomy of image use. Such a taxonomy has been developed within the VISOR project [8], set up to address the issue of how professionals search for, retrieve and use image data. The project aimed to build a robust model of information seeking behaviour. Forty-five users of manual and digital retrieval systems from 10 organisations participated in the study. Introductory meetings with participants were followed by informal observation of working environments, elicitation of verbal protocols from participants engaged in image-searches, questionnaires covering demographics and organizational issues, and finally individual in-depth interviews which focussed upon organization structure, the individual's role and their interaction with others, the search process and their use of image storage and retrieval systems.

Participant profiles revealed wide differences in image-seeking behaviour between different users. However it was possible to identify 7 different classes of image use:

***Illustration***, where images are used in conjunction with some form of accompanying media (e.g. news images)

***Information processing***, where the data contained in the image is of primary importance (e.g. X-rays)

***Information dissemination***, where the information in an image must be transmitted to someone else (e.g. mug-shots sent to police officers)

***Learning***, where people gained knowledge from image content (e.g. photographs or artwork used in academic research)

***Generation of ideas***, where images are used to provide inspiration or provoke thought (e.g. architecture, interior design images)

***Aesthetic value***, where images are simply required for decoration

***Emotive***, where images are used for their visceral impact (e.g. advertising).

In our present study, this taxonomy of image use was combined with the taxonomy of image content presented in section 1.2 above, to structure questions about those interface elements that users believed should be made available to support the process of image query.

## 2 Methods

A questionnaire was developed to gather data about the extent to which participants were interested in different types of content and the extent to which different content categories were preferred for different uses. It had four main sections:

- Rating the importance of each type of content in typical usage.
- Rating the usefulness of suggested interface elements for a particular query.
- Questions rating the perceived usefulness and ease of use of four example interaction styles.
- Personal information, including experience with image databases.

Participants were recruited through two main channels: direct requests to participants in previous studies and other contacts, and electronic mailing lists with archival or image data concerns. Emails asked recipients to forward details of the questionnaire to colleagues in an attempt to 'snowball' recruitment.

Logs showed a fairly low, but not problematic, completion rate. From the 668 who reached the introductory screen of the questionnaire, only 125 usable questionnaires (30.2% of those who began) were received.

## 3 Results

### 3.1 Respondent and Usage Profile

Of the 125 participants, 69 were female, 52 male, and four undeclared. The vast majority of participants fell into one of three age ranges: 45 were aged 25-34, 51 were aged 35-49, and 20 were aged 50-64. The most popular employment categories were museums or galleries (33), higher education (25), research (16), dedicated image libraries (10) or the media (9). Of the 27 participants who answered 'other' to this question, most mentioned some sort of library or archive, suggesting that the vast majority of participants worked in the public sector. Most participants were frequent users of image databases. On a scale of 1-7, where 1 indicated they never used a database and 7 indicated they used one every day, their mean rating was 5.38.

### 3.2 Importance of Each Category

Respondents were asked to rate the importance of each of the types of content identified in our earlier work (see Section 1.2) to the type of search they normally carried out, using a seven-point scale where 1 indicated no importance and 7 high importance. Results are shown in columns 2-4 of Table 1, overleaf.

A one-way repeated analysis of variance showed that importance ratings taken across all respondents differed significantly between categories ( $F(6.278, 703.090) = 37.137, p < 0.001^1$ ), with specific semantic terms the highest-rated category and emotional abstraction the lowest rated. In general, categories reflecting what Eakins and Graham [2] have described as level 2 content (semantic terms, cultural and

---

<sup>1</sup> Mauchley's test of sphericity indicates that a correction needs to be applied to these figures, due to the low sample size. The Huynh-Feldt epsilon correction was therefore applied to the number of degrees of freedom.

technical abstractions) received the highest ratings, with categories reflecting level 1 content (shape, colour, and texture) considered less important, and level 3 content (emotional abstraction) least important of all.

**Table 1.** Importance of each type of visual query element

	Whole population			Those actually using feature in searches		
	N	Mean	S. D.	N	Mean	S. D.
Semantic (specific)	123	6.16	1.32	104	6.39	1.03
Semantic (general)	121	5.69	1.58	104	5.89	1.50
Sharpness	122	5.20	1.93	55	6.15	1.05
Cultural abstraction	123	5.14	1.76	45	5.84	1.17
Technical abstraction	122	4.60	1.94	45	5.98	1.06
Metadata	123	4.26	1.99	56	5.11	1.67
Contextual abstraction	122	4.21	1.89	38	5.47	1.58
Colour	123	3.79	2.24	30	5.59	1.62
Shape	123	3.67	1.97	22	5.05	1.43
Texture	123	3.66	2.08	14	5.46	1.66
Visual relationships	122	3.57	1.89	23	4.96	1.46
Visual extension	120	3.56	1.87	23	4.95	1.50
Emotional abstraction	122	3.04	1.87	12	5.27	1.56

**3.3 Usage of Different Types of Image Content**

In order to explore use of different types of retrieval cue in more depth, participants were asked to provide a specific example of a recent query. Examples included "A spring image of one of the abbeys in the Scottish Borders", "Manuscript letters showing signatures" and "Aerial photographs of gasworks". Participants were also asked to indicate whether they actually used each particular type of image content in searching. The frequency of use of each type of image content is shown in column 5 of Table 1. Spearman's rank correlation between frequency of use and mean importance was highly significant ( $R = 0.909$ ,  $p < 0.001$ ), indicating that the more important a feature was rated, the more likely it was to be used.

Columns 6-7 of Table 3 show importance scores for each types of image content for those participants who had actually used them. Not surprisingly, mean scores were uniformly higher for users than non-users, though the difference was much more marked for some categories than others. The biggest rises in perceived importance were seen in the relatively specialized categories of technical and emotional abstraction, followed by categories characterizing image appearance, such as colour, texture and shape. The smallest rises (even when allowance is made for the limited scope for improvement in their scores) were seen with semantic terms and metadata.

### 3.4 Usage Within Each Category of Image Content

Those who used a given type of image content were then asked to indicate (again on a 7-point scale) how useful they expected different ways of specifying this type of content to be in answering their query. For example, participants who indicated that colour was important to them were asked follow-up questions about the usefulness of colour palettes, colour spectrum chart, and colour layout charts in their query.

**Colour** was considered a much more important search feature by those using it than respondents as a whole. However, mean usefulness ratings for the individual interface elements proposed for colour searching were uniformly low, ranging from 3.6 out of a possible 7 for *Colour Spectrum* down to 3.1 for *Colour Layout*.

**Sharpness**, by contrast, was considered important by the whole sample, not just by those who used it. This is perhaps a surprising result, as few current systems offer this as a search criterion.

**Texture** was not considered particularly important by participants overall, a rating reflected in its low usage - although those who did use it rated its importance much more highly. Mean usefulness ratings for individual methods of specifying texture were generally quite high, ranging from 5.1 for *Select a natural texture* to 4.5 for *Draw a texture freehand*.

**Shape**, like texture, was not considered particularly important by the whole sample, but was rated as important by the minority who did use it. Mean usefulness scores for individual methods of specifying shape were higher than for colour, but lower than for texture, ranging from 4.3 for *Select shape from list* through 3.5 for *Draw shape freehand* and 3.4 for *Select geometric shape*.

**Visual relationships** and **visual extension** received low usefulness ratings, both from respondents as a whole and from those who indicated that they had used them.

**Semantic content** was the most commonly used category, with 104 participants reporting usage of both general and specific semantic terms. Both types of term were rated as highly useful. All options for posing semantic queries were considered to be useful, with mean usefulness scores ranging from 6.5 for *Type a specific semantic term* down to 5.1 for *Semantic QBE*<sup>2</sup>.

**Contextual abstraction** was not rated as particularly important overall, though the sub-sample using it rated its importance more highly. Mean usefulness ratings of individual methods of specifying queries differed significantly, from 6.2 for *Type a contextual term* to 5.2 for *Context QBE*.

**Cultural abstraction** was considered important by a relatively large number of our respondents, both users and non-users. Mean usefulness ratings of individual methods of specifying queries again varied significantly, from 6.5 for *Type a cultural term* to 5.5 for *Select a cultural term from list*.

**Technical abstraction** was used moderately often, and given moderate importance ratings. Significant differences were found between individual methods of formulating such queries, with *Typing in a query* receiving the highest mean usefulness rating (6.0) and *Use graphical template* prompt the lowest (5.4).

**Emotional abstraction** appeared to be the least used, and least important, of the categories - though again, those participants who did use this category gave it higher importance scores. Differences between individual ways of expressing emotional

---

<sup>2</sup> Query By Example

queries (such as *Select emotional term from list* and *Emotion QBE*) were not significant - all had mean usefulness scores between 5.0 and 5.3.

*Metadata* was the second most frequent category of data used, though its importance ratings were not particularly high. Image type (whether photographic, painting, or scan) appeared the most important individual type of metadata (mean usefulness score 6.2), and aspect ratio the least (mean usefulness score 4.7).

3.5 Interaction Style

The questionnaire presented all participants with four alternative designs for the overall interaction, each illustrated with a small static image:

- All controls presented on a single screen - an entire query can be entered at once;
- Controls separated into groups on different screens, which can be accessed through tabs on the screen;
- Controls on separate screens, which must be completed in a fixed order;
- User provides rough sketch of query, adding colour & texture to specific objects.

These were rated on two scales derived from Davis' technology acceptance model [9] - perceived usefulness and perceived ease of use (table 2).

Table 2. Preferences for interaction style

		N	Min	Max	Mean	S.D.
Interface 1 – All at once	Usefulness	118	1	7	5.13	1.76
Interface 1 – All at once	Usefulness	118	1	7	5.13	1.76
	Ease of use	119	1	7	4.51	1.75
	Ease of use	119	1	7	4.51	1.75
Interface 2 – Tabbed	Usefulness	119	1	7	4.71	1.58
Interface 2 – Tabbed	Usefulness	119	1	7	4.71	1.58
	Ease of use	119	1	7	4.30	1.61
	Ease of use	119	1	7	4.30	1.61
Interface 3 – Sequential	Usefulness	118	1	7	3.31	1.59
Interface 3 – Sequential	Usefulness	118	1	7	3.31	1.59
	Ease of use	119	1	7	3.29	1.67
	Ease of use	119	1	7	3.29	1.67
Interface 4 – Sketch based	Usefulness	117	1	7	2.89	1.85
Interface 4 – Sketch based	Usefulness	117	1	7	2.89	1.85
	Ease of use	117	1	7	2.91	1.64
	Ease of use	117	1	7	2.91	1.64

Only the first two alternatives had mean usefulness or ease of use ratings higher than the scale midpoint. This suggests that an interface along the lines of alternative 1 is more likely to be used, a conclusion supported by asking participants to express a simple preference. The dominant first choice was alternative 1, with a moderate strength of preference (mean 4.91, SD = 1.72).

## 4 Discussion

### 4.1 Usefulness and Importance of Query Elements

The questionnaire asked participants to rate the importance of different types of content in their work, and the usefulness of different specific interface elements in query formulation. There was general agreement among our respondents that the most important types of content were semantic terms (either general, describing desired types of object or scene, or specific, naming individual objects or people), sharpness, and cultural and technical abstraction. This indicates clearly that the majority of our participants were interested primarily in what the pictures depicted - or, using Enser's terminology [10], that they were more interested in concept-based retrieval than content-based retrieval. This is an unsurprising finding, and reinforces those of earlier studies [4].

The preferred method for specifying such queries to a retrieval system was to type search terms in at a keyboard (respondents were not asked about voice input, though this could be seen simply as another way of specifying the same type of search term). Alternatives such as selection from a hierarchy of terms, or query by example, were consistently rated lower. This suggests that when semantic concepts are important in a search, direct access to those semantic terms may be more important than following a system-defined structure. This preference for typing a selection seems to go against the perceived simplicity of using a visual point-and-click interface.

The appearance of sharpness near the top of the ranking list is perhaps one of the most surprising findings of this study, since it is seldom provided as an explicit search criterion. Types of content relating to image appearance (colour, texture and shape) were given low importance ratings by respondents as a whole, but did get moderately high rankings from those who actually used them in searching. This again accords with evidence from Enser [4] and others, suggesting that retrieval by appearance is of interest only to a minority. Retrieval by colour appeared to be more important than retrieval by texture or shape. But even for those who had used it, colour was ranked only sixth in order of importance, behind semantic concepts and cultural and technical abstraction.

Users' ratings of the usefulness of individual methods of specifying primitive-level queries were generally low. No method of specifying colour came above the mid-point ranking, and only one method of specifying shape. This dissatisfaction with current methods of specifying desired colours might be a reason for the low usage of colour among our respondents. Interestingly, the few users who searched by texture seemed to rate specification methods more highly - though none of the differences between specification methods reached statistical significance. Users did however seem to express a slight preference for selecting elements such as shape or texture from a menu, rather than trying to enter them freehand. Further evidence to support this conclusion comes from Table 2, where a sketch-based interface was rated least useful of the four alternatives presented.

The low usage of emotional content was striking. This may have been due to lack of awareness of its potential among respondents as a whole. Alternatively, searchers may have rejected an emotional expression of their query because they did not expect the system to be able to interpret it adequately. It is noticeable that the minority who did use emotional content ranked its importance relatively highly, and appeared to consider all methods of specifying it relatively useful.

4.2 Implication for Retrieval Systems Design

Table 3 lists the six individual interface elements with the highest usefulness ratings. It is interesting that of these six elements, four are based on simple text entry. This is perhaps not encouraging for developers of novel interfaces to image database systems, nor to those developing CBIR systems. One of the potential benefits claimed for CBIR is that it removes, or at least reduces, the need for linguistic coding of complex content, and eases international applications by removing linguistic barriers. This does not appear to be borne out by our study.

**Table 3.** Methods of specifying image content which received usefulness ratings above 6

Type of content	Interface element	N	Mean	S.D.
Semantic units	Type specific semantic term	103	6.51	1.14
Cultural abstraction	Type cultural terms	44	6.45	.79
Metadata	Image type (photo, painting)	54	6.20	1.41
Emotional abstraction	Emotion polarity	10	6.20	.92
Contextual abstraction	Type contextual term	35	6.08	1.24
Technical abstraction	Type technical term	44	6.02	1.34

User responses suggest that an ideal interface to an image retrieval system should provide a single screen containing all necessary controls, so that an entire query could be entered at once. The range of interface components provided should include the following:

- free text entry (allowing entry of the names of desired objects, but also cultural, contextual and technical terms. These may be a single field, or separated);
  - specification of overall focus and sharpness of image;
  - selection of the type of original image required (e.g. photograph, painting);
  - selection of image file criteria (size and type);
  - whether the image has positive or negative emotional impact;
  - menu selection of technical terms (where appropriate);
- though it would probably be advantageous to allow users to customize this screen to their individual requirements.

Most of these could readily be supported by existing technology, provided human indexers were available to label the images with appropriate semantic content (including contextual and technical terms). Many current image databases can automatically recognize different file types, and automatically extract metadata such as image height and width. CBIR techniques have already been successfully applied to the problem of automatic recognition of image type [11], and analysis of an image's Fourier spectrum should give an adequate indication of its sharpness. The ability to recognize emotional polarity without specific human input represents more of a challenge.

Several possible factors could explain users' apparent preferences for simple text input over alternatives such as menu choice or query by example. Our respondents had a high level of overall experience of using image databases, so may simply have been expressing a preference for methods with which they were already familiar. Another factor may have been users' desire to maintain independence and control, and not have a system impose a structure on to their thinking. Typing a query allows more



freedom than selecting terms from a list. Evidence to support this comes from users' declared preference to have the whole query interface available at one time rather than having a system-defined sequence for their query. Selection of appropriate methods for formulating visual queries (such as those involving colour or shape) seems to be more problematic. As indicated above, none of the methods of visual query formulation suggested in our questionnaire found widespread favour with our respondents.

## 5 Conclusions

The ability to retrieve images by their semantic content is a clear priority for users of image databases. Lower level issues are generally considered less important. Users with experience of image databases may have more conservative demands and expectations of a system's ability to deliver in response to higher level queries. System design must take account of this, while trying to introduce newer, more efficient interaction to those users. Our study reinforces that view that current interfaces to CBIR systems are inadequate for user needs, and suggests that the time is ripe for investigating new query paradigms.

**Acknowledgements.** The financial support of the Arts and Humanities Research Board is gratefully acknowledged.

## References

1. Bjarnestam, A (1998) "Description of an image retrieval system", presented at *The Challenge of Image Retrieval* research workshop, Newcastle upon Tyne, 5 February 1998
2. Eakins, J P and Graham M E (1999) Content-based Image Retrieval: A report to the JISC Technology Applications Programme [<http://www.unn.ac.uk/iidr/report.html>]
3. Smeulders AWM, Worring M, Santini S, Gupta A & Jain R (2000) Content-based image retrieval at the end of the early years. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 22(12), 1349-1380
4. Enser P G B (1995) Pictorial Information Retrieval *Journal of Documentation* 51: 126-170
5. Eakins, J P (2002) "Key issues in image retrieval research" Keynote address to ASCI2002 conference, Lochem, the Netherlands
6. Flickner, M et al (1995) "Query by image and video content: the QBIC system" *IEEE Computer* 28(9), 23-32
7. Burford, B, Briggs, P & Eakins J P (2003) A taxonomy of the image: on the classification of content for image retrieval *Visual Communication* 2(2), 123-161
8. Conniss, L R, Ashford, J A and Graham, M E (2000). *Information seeking behaviour in image retrieval: VISOR I* Final Report. Institute for Image Data Research (Library and Information Commission Research Report 95).
9. Davis, F D (1989) Perceived usefulness, perceived ease of use, and user acceptance of information technology. *I* 13(3), 319-340
10. Enser, P G B (2000) Visual image retrieval: seeking the alliance of concept-based and content-based paradigms. *Journal of Information Science* 26(4) 199-210
11. Athitsos V, Swain M J and Frankel C (1997) Distinguishing photographs and graphics on the World Wide Web. *Proc IEEE Workshop on Content-based Access of Image and Video Libraries*, 10-17

# SCULPTEUR: Multimedia Retrieval for Museums

Simon Goodall<sup>1</sup>, Paul H. Lewis<sup>1</sup>, Kirk Martinez<sup>1</sup>, Patrick A.S. Sinclair<sup>1</sup>,  
Fabrizio Giorgini<sup>2</sup>, Matthew J. Addis<sup>3</sup>, Mike J. Boniface<sup>3</sup>, Christian Lahanier<sup>4</sup>,  
and James Stevenson<sup>5</sup>

<sup>1</sup> Electronics and Computer Science, University of Southampton, SO171QP, UK.  
`{sg,phl,km,pass}@ecs.soton.ac.uk`

<sup>2</sup> Giunti Interactive Labs, Sestri Levante, 16039, Italy. `f.giorgini@giuntilabs.it`

<sup>3</sup> IT Innovation Centre, Southampton, SO167NP, UK.

`{mja,mjb}@it-innovation.soton.ac.uk`

<sup>4</sup> Centre de Recherche et de Restauration des Musées de France, Paris, France.

`Christian.Lahanier@culture.gouv.fr`

<sup>5</sup> Victoria and Albert Museum, South Kensington, London. `J.Stevenson@vam.ac.uk`

**Abstract.** The paper describes the prototype design and development of a multimedia system for museums and galleries. Key elements in the system are the introduction of 3-D models of museum artefacts together with 3-D as well as 2-D content based retrieval and navigation facilities and the development of a semantic layer, centred on an ontology for museums, which aims to expose the richness of knowledge associated with the museum collections and facilitate concept based retrieval and navigation integrated with that based on content and metadata. Interoperability protocols are designed to allow external applications to access the collection and an example is given of an e-Learning facility which uses models extracted to a virtual museum.

## 1 Introduction

Museums, galleries and other cultural heritage institutions are finding it beneficial, to maintain and exploit multimedia representations of their collections using database indexing and retrieval technology. Their requirements present researchers with a growing series of challenges, particularly as the range of representations is expanding to include, for example, 3-D models and digital videos and the technologies available are evolving rapidly, notably in the area of web services and the semantic web.

This paper provides a progress report on a project which is designed to meet some of these new challenges, especially in the area of retrieval and navigation of multimedia information in the context of emerging semantic web technology. The project, SCULPTEUR[2,1], involves five major European galleries, the Uffizi, the National Gallery and the Victoria and Albert Museum in London, the Musée de Cherbourg and the Centre de Recherche et de Restauration des Musées de France (C2RMF) which is the Louvre related art restoration centre. Each of

these maintains a substantial digital archive of its collections. Other technical partners include Centrica in Italy and GET-ENST in Paris. The project builds on the work of an earlier museum database project, ARTISTE[11,3].

One of the goals of SCULPTEUR is to extend the retrieval and navigation facilities of the digital museum archives to 3-D multimedia objects. Increasingly museums are recognising the value of 3-D visualisation of their artefacts not only for researchers, curators and historians but also potentially as an information source for a wider public and as a basis for e-learning and commercial activity. A second new goal of the project is concerned with the perceived benefit of structuring and integrating the knowledge associated with the museum artefacts in an ontology. Metadata associated with the artefacts is being mapped to the ontology to form an integrated knowledge base and graphical tools are being developed to provide browsing of the concepts, relationships and instances within the collections. Integrated concept, metadata and content based retrieval and navigation facilities are being implemented to explore the knowledge base. Other goals include the development of a web agent to locate missing metadata and also exploitation of the system by development of an e-learning product to make use of it.

In section 2 related work is presented and in section 3 we describe some of the specific needs of our users, identified at the start. Sections 4 and 5 discuss issues relating to building the ontology based architecture and multimodal, multimedia retrieval respectively. Section 6 presents an example of interoperability using an e-Learning example and finally, in section 7, some conclusions and challenges are described.

## 2 Related Work

The SCULPTEUR project is related to and builds on previous work in several fields. A major source of inspiration comes from previous work on the ARTISTE and MAVIS projects[10,11] and a large body of other published work on content based image retrieval systems[12]. 3-D model capture is an important element and one of the partners, GET-ENST, has developed techniques for accurately generating 3-D models from multiple views[19]. Various authors have published algorithms for 3-D model matching using a variety of feature vectors extracted from mesh based 3-D representations based on for example, 3-D Hough transforms[18], 3-D moments[17] and surface features such as chord distributions[14] and radial axis distributions[15]. The work from Princeton has been particularly influential in this area and a recent paper on 3-D benchmarking compares a range of matching algorithms[16] in terms of retrieval performance using the Princeton Benchmark data set. The idea for the semantic layer and knowledge base draws on previous work at Southampton[20] and semantic web technology[6]. Various other groups have reported the use of ontologies for image annotation.

### 3 Identifying User Needs

The ARTISTE project combined metadata based retrieval with both general purpose content based retrieval tools such as colour, spatial colour and texture matching with specialist content based facilities to meet some specific needs of the participating museums. These included, for example, sub-image location[21], low quality query image handling[22] and canvas crack analysis and retrieval[9]. The ARTISTE system was the starting point for our work in SCULPTEUR and the new needs of the museums were identified, particularly in relation to their increasing use of 3-D digital representations of their artefacts. These included the ability to compare and retrieve 3-D objects on the basis of size, colour, texture and 3D shape. Examples of more specialised requirements included the ability to retrieve objects by sub-parts and by detail measurements, the ability to classify objects using these features and the ability to correlate for example figurines and moulds in which they may have been made by the external and internal 3-D profiles respectively. All these requirements and others relating to the forms of retrieval and navigation required and the ability to integrate with external applications such as e-Learning packages, impose substantial demands on the architecture and functionality required in the new system.

### 4 Building the Knowledge Base

The current SCULPTEUR architecture is shown in figure 1. The system is implemented as a web based client-server application and the server-side repositories hold the raw multimedia objects, feature vectors for content-based retrieval, textual metadata and the ontologies which are implemented using Protégé[8] and held as RDF[13]. Tools have been developed for importing new or legacy objects or mapping directly to existing collections in situ. Above the repositories, components of the architecture provide semantic integration between concepts and relations in the ontology and metadata and media objects in the repositories plus search, browsing and retrieval services for both SCULPTEUR driven activity and eventually for externally invoked activity via the interoperability protocols, SRW[5] and OAI[4]. The users' desktop is essentially a standard browser augmented with 2-D and 3-D model viewers, facilities for query formulation such as colour pickers, and uploading facilities for query models.

The starting point for the development of the ontology was the conceptual reference model (CRM)[7] developed by the documentation standards committee of the International Council of Museums (CIDOC). The aim of the CRM is to support the exchange of relevant cultural heritage information across museums, based on shared semantics and common vocabularies. Working closely with the museums is necessary to extend the core CRM to enhance the particular areas relevant to each museum and to develop mappings between museum metadata values and concepts in the ontology. The problems and difficulties of developing the semantic layer in this way should not be underestimated. The importing and mapping of legacy museum data, both in terms of concepts and instances

in the ontology is a complex manual process involving collaboration between technologists, domain experts and CRM experts. Problems occur at various levels from establishing coherent semantics at the highest level to interpreting or unravelling obscure coding and formats at the lowest. Tools are being developed to use the mappings to automatically build associations between media objects and concepts, making them instances of the concepts in the ontology.

Using the ontology to develop a semantic layer in this way, provides a bridge across the semantic gap, facilitating search for media objects via the concepts and relationships in the ontology as well as via more usual content and metadata based searching. In addition, the semantic layer aims to expose the richness of information surrounding the media objects themselves, allowing the system to move from one in which search is solely focussed on the media objects to one in which any of the entities in the semantic layer can become the focus of the investigation. Thus, the system will not only allow queries of the type "Find me all 3-D objects in the collection with a shape similar to this query object" but also queries focussed on other entities such as "Find me all countries that produced artists working in the 17th Century."

The architecture also includes a system ontology which captures the concepts and relationships associated with the system itself. The system consults the ontology to determine which tools and components to use for a particular task and will eventually be able to expose its facilities more elegantly to external agents wishing to use them.

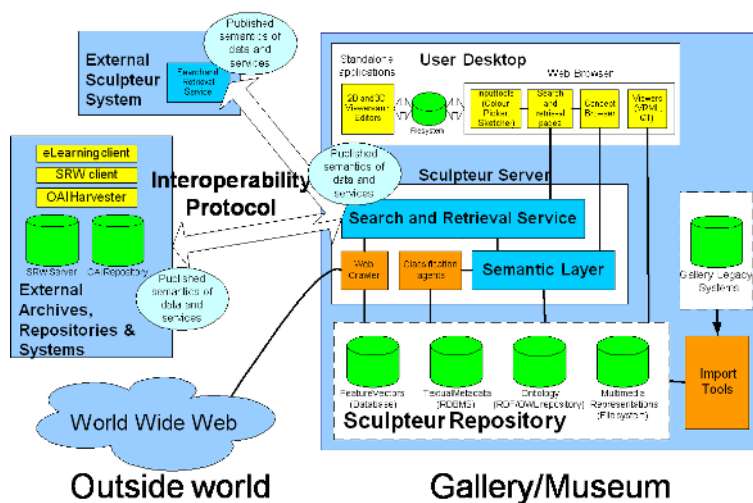


Fig. 1. The SCULPTEUR Architecture

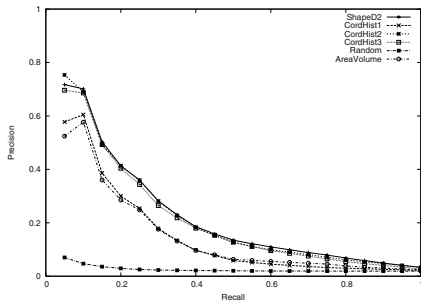
## 5 Multimodal Multimedia Retrieval

In addition to the 2-D content-based image retrieval (CBR) tools of ARTISTE, the prototype SCULPTEUR system is designed to support content-based retrieval of 3-D object models by model matching. In one sense 3-D content-based retrieval is more straight forward than 2-D as the objects are explicitly represented rather than embedded in a pixel (or voxel) matrix. However, mesh based representations are not immediately amenable to comparison for CBR applications and 3-D CBR techniques compare the similarity of the shape of the object as represented by some feature vector extracted from the mesh.

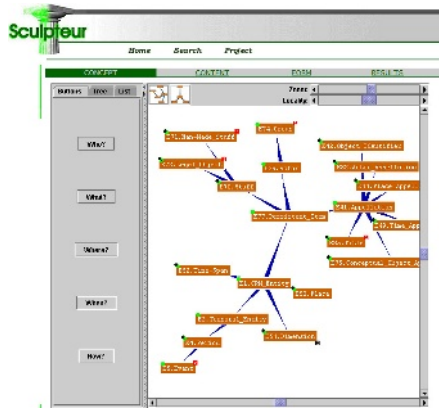
So far in SCULPTEUR several published 3-D matching algorithms have been implemented and integrated with retrieval facilities in the first prototype to provide 3-D CBR. These include the D2 shape distribution descriptors from the Princeton Shape Retrieval and Analysis Group[14], the histogram descriptors from Paquet and Rioux[15] and the Area to Volume Ratio descriptor[23] which is a single statistic giving the ratio of the surface area of the model to its enclosed volume. The D2 descriptor records the distribution of distances between random points on the surface of the model and is rotation and translation invariant and robust to changes in mesh resolution. In our implementation, a 64 bin histogram was used for D2. There are three versions of the histogram descriptors of Paquet and Rioux . They define a cord as the vector between the centre of mass of an object and a point on its surface. Their first histogram records the distribution of the cord lengths for all points within the mesh. The other two variations record the distribution of angles between cords and the first and second principal axis respectively. Each of these was implemented as a 16 bin histogram. Before including them in the prototype, an evaluation was made of the five 3-D algorithms for CBR. Using the Princeton Benchmark[16] base dataset, (training group) consisting of 907 models representing about 90 object classes, precision-recall graphs of the five algorithms were created. They are shown in figure 2 together with the precision-recall graph for random retrieval. It can be seen that the D2 descriptor gives the best retrieval results in terms of precision-recall and as expected, the more basic Area to Volume Ratio descriptor gives the poorest retrieval results. The three Paquet and Rioux histograms are in between but the histogram of the angle between the cord and principal axis version gives the best retrieval results of these three.

An example of retrieval results in SCULPTEUR is shown in figure 4. The top ten best matches are shown for a 3-D retrieval using a vase as the query and the D2 descriptor for matching. The first match is the query object as expected. The test dataset used here consists of around 300 models from both the museum partners and from other collections.

The concept browser, shown in figure 3, provides graphical navigation of the semantic layer. Concepts in the ontology are represented by the nodes in the graph and the relationships between concepts are represented by the graph edges. In the current prototype, the feature vectors are only associated indirectly with concepts via the media objects, but the interface does allow the combination of concept based retrieval with the other retrieval modes.



**Fig. 2.** Precision/Recall for shape descriptors



**Fig. 3.** Concept Browser



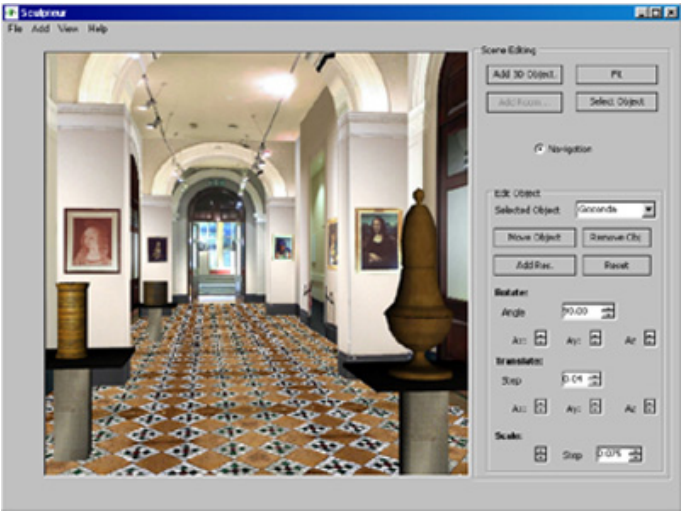
**Fig. 4.** Top ten results for a query vase (first image is also the query)

Certain metadata fields contain complex or non-atomic values that express relationships between records, and the concept browser is able to display these fields with the graphical interface in a new window.

## 6 Interoperability: An E-learning Application

One of the goals of the SCULPTEUR system is to provide an easy interface to external applications which may wish to utilise the media objects or other knowledge in the system. As an example of such an application, Giunti, Interactive Labs, the project coordinators, are integrating their e-Learning package using the interoperability protocol, SRW, implemented in the system. The integration is motivated by the recent increased interest by cultural institutions in reusable multimedia components for learning (called Cultural Learning Objects, CLO) and new technologies, capable of online learning contents delivery and management. The result is a content authoring tool able to create and manage 3D virtual learning environments of Cultural Learning Objects.

The user interface for the Giunti system, shown in figure 5, assists curators, instructional designers and educators to build virtual exhibitions of 3-dimensional Cultural Learning Objects, define learning paths and package the 3D virtual learning environment according to the new e-learning specifications defined by IMS (Instructional Management System[24] which eases the exchange of cultural contents and educational material from one museum to another.



**Fig. 5.** User interface for the Giunti e-Learning System

According to the defined learning path, the tool automatically generates the SCORM run-time environment APIs[25] for the tracking of users' actions and the communication with a Learning Management System and embeds these calls in the generated VRML. This allows a Learning Management System to control the learning paths on-the-fly in accordance with the experiences made by the user while he/she navigates the 3D virtual environment. In order to establishing an appropriate virtual environment the e-Learning package will query the SCULPTEUR system for appropriate artefacts to place in the virtual museum and these will be delivered through the SRW interface.

## 7 Conclusions and Some Outstanding Challenges

The paper has presented a prototype multimedia system which we are in the process of developing. The project aims to capitalise on emerging semantic web technologies and novel 3-D retrieval to provide museums and galleries with more versatile facilities for exploring and exploiting their digital collections. The project is on-going and several major challenges remain.



The ontology, at the heart of the semantic layer, serves a number of purposes. Notably, it provides a basis for interoperability between digital libraries and, for example, with e-learning facilities. Through the concept browser, it also aims to provide a high level navigation and retrieval interface for the collections. However, to be a realistic shared conceptualisation of the museum domain, the ontology is of necessity, a large and complex representation. One of the significant challenges is to make a more intuitive, easy to use interface for exploration and navigation which still exposes the richness of the collections. The ontology also needs closer integration with the content and metadata based retrieval to provide enhanced retrieval capabilities, for example by automatic query expansion through the semantic layer. We have also developed a prototype web crawler which seeks for missing information from the ontology on the web. In the absence of widespread uptake of semantic web technology, the information extraction process is mainly via natural language processing and many problems and opportunities for improvement in functionality remain here.

The first prototype of SCULPTEUR is currently under evaluation by the five galleries involved in the consortium. Feedback from this process will enable us to continue to evolve the system towards a more useful exploration and retrieval facility for museum collection management.

**Acknowledgements.** The authors wish to thank: the European Commission for support through the SCULPTEUR project under grant IST-2001-35372. Thanks also to our collaborators including F. Schmitt and T. Tung of GET-ENST, R. Coates of the V&A museum, J. Padfield of the National Gallery, R. Rimaboschi of the Uffizi, J. Dufresne of the Musée de Cherbourg and M. Cappellini of Centrica for many useful discussions, use of data and valuable help and advice; Patrick Le Boeuf of the Bibliothèque Nationale de France for assistance with mapping to the CRM; TouchGraph ([www.touchgraph.com](http://www.touchgraph.com)) for software used in the concept browser; Hewlett Packard's Art & Science programme for the donation of server equipment, the ARCO Consortium[26] for the VRML model of the VAM Art Decò corridor and other models.

## References

1. Addis, M., Boniface, M., Goodall, S., Grimwood, P., Kim, S., Lewis, P., Martinez, K., Stevenson, A.: SCULPTEUR: Towards a new paradigm for multimedia museum information handling. 2nd International Semantic Web Conference, pp 582–596, October 2003.
2. SCULPTEUR IST-2001-35372 <http://www.sculpteurweb.org>
3. ARTISTE IST-1999-11978 <http://www.artisteweb.org/>
4. Open Archives Initiative <http://www.openarchives.org/>
5. ZING Search and Retrieve Web service  
<http://www.loc.gov/z3950/agency/zing/srw/>
6. The Semantic Web <http://www.semanticweb.org/>
7. Crofts, N., Dionissiadou, I., Doerr, M., Stiff, M.: Definition of the CIDOC Object-Oriented Conceptual Reference Model, V.3.1. July 2001

8. Gennari, J., Musen, M., Fergerson, R., Grosso, W., Crubzy, M., Eriksson, H., Noy, N., Tu, S.: The Evolution of Protégé: 2002. Technical Report, SMI-2002-0943
9. Abas, F., and K. Martinez: Craquelure Analysis for Content-Based Retrieval. IEEE DSP, pp 111–114, 2002 conference. July 2002.
10. Dobie, M. R., Tansley, R. H., Joyce, D. W., Weal, M. J., Lewis, P. H., Hall W.: A Flexible Architecture for Content and Concept Based Multimedia Information Exploration. Proc. The Challenge of Image Retrieval, Newcastle, 1999, pp 1–12.
11. Addis, M., Boniface, M., Goodall, S., Grimwood, P., Kim S., Lewis, P., Martinez, K., Stevenson, A.: Integrated image content and metadata search and retrieval across multiple databases, Proc. of Second Int. Conf. on Image and Video Retrieval 2003, pp. 91-100, Illinois, USA.
12. Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., Jain. R.: Content-Based Image Retrieval at the End of the Early Years, PAMI volume 22 of 12, pages 1349-1380 2000.
13. RDF Resource Description Framework <http://www.w3c.org/RDF/>
14. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D models with shape distributions, in Proc. Shape Modelling International, IEEE Press, Genova, Italy, 2001, pp. 154–166.
15. Paquet, E., Rioux, M.: Nefertiti: a Query by Content Software for Three-Dimensional Models Databases Management, in Proc.Int Conf. on Recent Advances in 3-D Digital Imaging and Modeling, May 12-15, 1997, pp. 345–352.
16. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The Princeton Shape Benchmark, to appear in Shape Modeling International, Genova, Italy, June 2004
17. Elad, M., Tal, A., Ar., S.: Content Based Retrieval of VRML Objects - An Iterative and Interactive Approach. Proceedings of the sixth Eurographics workshop on Multimedia 2001. Manchester, UK, pp. 107–118
18. Zaharia, T., F.Preteux, F.: Hough transform-based 3D mesh retrieval Proceedings SPIE Conference 4476 on Vision Geometry X, San Diego, CA, August 2001, pp. 175–185.
19. Esteban, C. E., Schmitt, F.: Silhouette and Stereo Fusion for 3D Object Modeling. Fourth International Conference on 3-D Digital Imaging and Modeling , pp. 46–54 October 06-10, 2003, Banff, Alberta, Canada.
20. Lewis, P. H., Davis, H. C., Dobie, M. R. and Hall, W. : Towards Multimedia Thesaurus Support for media-based Navigation. In Proceedings of Image Databases and Multi-Media Search, Series on Software Engineering and Knowledge Engineering 8, pp 111–118. 1997
21. Chan, S., Martinez, K., Lewis, P., Lahanier, C. and Stevenson, J. : Handling Sub-Image Queries in Content-Based Retrieval of High Resolution Art Images,. Proceedings of International Cultural Heritage Informatics Meeting 2, pp. 157-163, 2001.
22. Fauzi, M. F. A. and Lewis, P. H. : Query by Fax for Content-Based Image Retrieval. Proceedings of International Conference on the Challenge of Image and Video Retrieval, LNCS vol. 2383, pp. 91-99, London, United Kingdom 2002.
23. C. Zhang and T. Chen, Efficient Feature Extraction for 2D/3D Objects: in Mesh Representation, ICIP 2001, Thessaloniki, Greece, pp 935–938, 2001.
24. IMS Global Learning Consortium, Content Package Specifications, <http://www.imsglobal.org/content/packaging/index.cfm>
25. Advanced Distributed Learning, Sharable Content Object Reference Model, <http://www.adlnet.org/index.cfm?fuseaction=AboutSCORM>
26. ARCO project, <http://www.arco-web.org>

# Disclosure of Non-scripted Video Content: InDiCo and M4/AMI

Franciska de Jong<sup>1,2</sup>

<sup>1</sup>University of Twente, Department of Computer Science,  
P.O. Box 217, 7500 AE Enschede, The Netherlands  
fdejong@ewi.utwente.nl

<sup>2</sup>TNO TPD, P.O. Box 155  
2600 AD Delft, The Netherlands

**Abstract.** The paper discusses three IST projects focusing on the disclosure of video content via a combination of low-level multimodal feature analysis, content abstraction, and browsing tools. The type of content (recordings of conference lectures and meetings) can be characterized as non-scripted and is argued to generate a whole range of new research issues. Performance results are reported for some of the tools developed in InDiCo and M4.<sup>1</sup>

## 1 Introduction

Since the mid nineties automated disclosure of video content has been on the agenda of several R&D projects within the European funding programmes. These projects have built on the insights and results obtained in fostering research at several laboratories and archiving institutes in Europe and elsewhere [1], [2]. As video material is a typical *multi-media* type of content, video indexing can in principle deploy and combine the analysis tools for various media. For video with a soundtrack two data channels are important, thus a first distinction is commonly made between audio and video. But to each of these two data types multiple modalities correspond. Audio may consist of speech, music and other sounds, either or not in combination. Video as it appears on a screen can also involve more than just images: often additional captions or subtitles come along, while shot images may contain textual elements as well. Most of these symbolic elements belong to the same realm as the speech parts in the audio: natural language, a medium that can help bridge the semantic gap between low-level video features and user needs, and that can also very well be exploited for the generation of time-coded indexes [3]. As for image content from the general domain no efficient and effective information semantic disclosure was nor is likely to become available soon, the deployment of speech and language processing tools played an important role in most early video retrieval projects ([4], [5]), and in the video retrieval evaluation conferences organised in the context of TREC ([6]).

---

<sup>1</sup> Thanks go to Mike Flynn and Dennis Reidsma for support in preparing this paper.

For obvious reasons (availability of test and training collections, performance expectations, commercial interest) the focus has long been on a specific type of pre-produced scripted content: broadcast news programmes. Evidently there are still a lot of problems to be solved in this domain, but due to both the boost of digital production and retrospective digitisation, an enormous growth in the number and size of digital video collections can be observed. A considerable part of this content can be characterized as non-scripted: recordings of events for which no script is available that strongly determines the behaviour of all registered people and objects. Examples are: logs of videoconferences, images captured by surveillance camera's, live reports of significant events from various domains (sports, celebrations, public ceremonies, etc.), and recordings of discussions, lectures and meetings. This paper will discuss three on-going IST-projects that have chosen recordings of non-scripted events as the target for content retrieval technology:

- project M4 (MultiModal Meeting Manager)  
<http://www.m4project.org/>
- project AMI (Augmented Multi-party Interaction)  
<http://www.amiproject.org/>
- project InDICO (Integrated Digital Conference)  
<http://indico.sissa.it/>

Though in each of these projects a number of different functionalities determine the research agenda, they all pay attention to the disclosure and retrieval of video content at the level of fragments, partly based on the use of language and speech processing to support a form of content abstraction and/or the automated generation of metadata. As the image content is relatively static, dominated by talking heads and moving bodies, not the indexing of video frames is the primary target, but the realisation of a truly multimedia retrieval environment in which all information modalities contribute to the disclosure of a media-archive up to a high level of granularity.

One of the interesting things about the non-scripted data collections is that they open up new types of usage that may require a much more diverse range of analysis and presentation techniques. The type of collections to be discussed here for example, are not just of interest for what they are about, but they also are a valuable source for researchers interested in the study and modelling of human interaction, both in terms of verbal and non-verbal behaviour. In order to allow researchers to use a meeting corpus for this aim, the analysis should not just focus on the image or speech content, but also on interpretation of multi-modal aspects like gaze, gesture, etc. New requirements will be posed on the fission of data, and it is likely that new designs are needed to enable content abstraction for data sets with an information density that is incomparable to what news archives have to offer. Likewise the presentation and ranking of retrieval results should facilitate search tasks that are not supported by the common tools for media fusion and browsing, partly because the content can be viewed from multiple perspectives. Therefore the inseminating role non-scripted data collections may play for the field of video indexing should not be underestimated.

This paper is organized as follows. In section 2 this paper describes the aims for each of the three projects at a general level. Section 3 will discuss the characteristics of non-scripted content and section 4 will present some of the results. The paper will be concluded by a discussion of how the performance of disclosure tools for non-

scripted video content may affect future research in the field of content-based video retrieval.

## 2 Projects' Aims

Both M4 and its follow-up project AMI focus on meeting recordings, while InDiCo aims at the disclosure of lecture recordings. The objectives vary considerably.

### 2.1 M4

The M4 project is concerned with the construction of a demonstration system to enable structuring, browsing and querying of an archive of automatically analyzed meeting recordings. The main focus is on a corpus of recordings of meetings that take place in a meeting room equipped with multimodal sensors (microphones, camera's) for a limited number (4) of participants, but some additional effort is put in the analysis of recordings of parliamentary sessions, with non-directed, natural behaviour and interaction.

In the case of recordings from the dedicated meeting room, data of a variety of types is generated and deployed in the analysis and disclosure of the meeting content. In addition to multiple audio channels and video streams from several cameras, there is additional information coming from interaction with PC's and an instrumented white board. Several tools for the segmentation of the audiovisual content are investigated. Experiments have been done with location-based speaker segmentation, and beamformed microphone array data. Visual processing focuses on the development of an audio-visual speaker tracker (which switches between speakers and works across cameras), face detection and tracking, and gesture recognition (e.g., pointing, standing up, sitting down.)

A meeting may be accessed by its structure (interaction patterns, dialog-acts, turn-taking, etc.), but also by what the participants say. The envisaged browsing facility that will make the content available via a media file server will initially focus on the latter, while the development for which some results are reported in section 4 is aimed at the former. To capture the structure a series of meeting actions has been defined (monologue, discussion, presentation, consensus, disagreement, ...) and models have been trained to automatically segment meetings in terms of these group actions, using audio features (such as speech activity, intonation, key words) and visual features (such as head detection). Cf. Figure 1 for a screen shot of an initial browsing prototype, displaying the structure of a discussion on favorite movies.<sup>2</sup>

### 2.2 AMI

AMI targets computer enhanced multi-modal interaction in the context of meetings. The project aims at substantially advancing the state-of-the-art, within important underpinning technologies (such as human-human communication modeling, speech recognition, computer vision, multimedia indexing and retrieval). It will also produce

---

<sup>2</sup> Development of the M4 browser was done at IDIAP, Martigny.

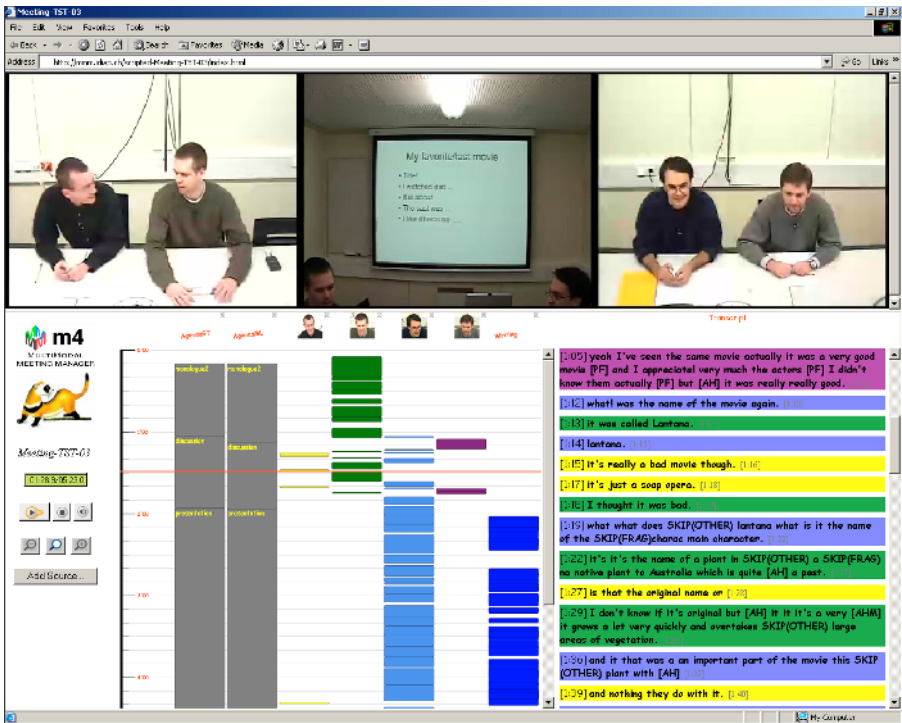


Fig. 1. Prototype of M4 browser

tools for off-line and on-line browsing of multi-modal meeting data, including meeting structure analysis and summarizing functions.

AMI performs the above research in the framework of a few well-defined and complementary application scenarios, involving an offline meeting browser, an online remote meeting assistant and integration with a wireless presentation system. These scenarios are being developed on the basis of smart meeting rooms and web-enhanced communication infrastructures. For the purpose of this paper the following three research areas are most relevant:

1. Multimodal low-level feature analysis, including multilingual speech signal processing (natural speech recognition, speaker tracking and segmentation) and visual input (e.g., shape tracking, gesture recognition, and handwriting recognition).
2. Integration of modalities and coordination among modalities, including (asynchronous) multi-channel processing (e.g., audio-visual tracking) and multimodal dialogue modeling.
3. Content abstraction, including multi-modal information indexing, cross-linking summarizing, and retrieval.

### 2.3 InDiCo

The objectives of the InDiCo project are to automate the process of managing conference content (papers, presentations, lecture recordings) for improved information sharing and exchange of the content via Internet. This is pursued by developing tools for the indexing and browsing of conference content, and to integrate these tools into an existing platform for digital publishing. Validation will be based on an experimental evaluation within the high-energy physics community at the CERN institute. Segmentation of lecture recordings into a segment-per-slide structure is a key technology. Additionally InDiCo developed domain specific speech recognition for non-native speakers and automatic clustering for the cross-linking of the content of conference speech transcripts, papers, slides and presentations, based on a memory-based learning classifier [9]. Eventually the result will be a novel navigational structure between video, slides and papers allowing users to combine the background of a paper with the compressed contents of a presentation.

## 3 Non-scripted Content Characteristics

In section 1, non-scripted content is described as recordings of events for which no script is available that strongly determines the behaviour of all participating people and objects. What this implies becomes immediately clear from the contrast with the characteristics of broadcast news programmes or movie scenes and other pre-produced and directed video recordings: what is said by whom is more or less prescribed (read speech), the position and movements of the ‘actors’ are heavily constrained, camera positions and turns follow a foreseeable pattern

Though in each of the three projects a number of different functionalities determine the research agenda, common goals are the development of tools for:

- indexing of video at the level of fragments, partly based on the use of language and speech processing
- content abstraction, to support efficient browsing through large volumes of archived content

So far there is a great overlap between research issues relevant for e.g., broadcast news archives. However, the difficulties to be solved differ because of the nature of the content and the envisaged applications. Here are two (not complete) lists of characteristics and requirements for the processing of two types of distinct non-scripted events, each corresponding with research issues that are absent or less dominant in the news domain:

Meeting recordings:

- low density of information
- gaze, gesture and movement may mark crucial events
- lack of training and evaluation material
- lack of evaluation methodology for the demonstrator functionality
- non-sequential speaker segmentation output due to overlapping speech

Conference recordings:

- ASR needed for non-natives

- lack of models for multimodal aspects of non-verbal presentation elements
- no fixed model of audience-speaker interaction
- domain models needed for highly specialised topics

Though incomplete and impressionistic, the two lists differ enough to illustrate that non-scripted multimedia content retrieval for these two domains can be viewed as two parameter instantiations, representing a more complex and a more simple case, respectively. Cf. Table 1 for a comparison of the varying parameter values for the *three* content domains and/or corresponding browse scenarios distinguished thus far.

**Table 1.** Varying parameter values for scripted and non-scripted content

	Pre-produced news broadcast	Meeting with $n$ participants	Lecture by 1 speaker, audience of $n$ people
a. Number of central speakers roles	1 or 2	2- $n$	1
b. Number of interactors per segment	2	2- $n$	2- $n$
c. Number of interaction moments	low	relatively high	relatively low
d. Number of cameras/ microphones	1 or 2	1 - $n$	1 or 2
e. Variation in speech characteristics (pronunciation, lexicon)	little	dependent of $n$	dependent of $n$ and (c)
f. Number of topics to be addressed	open	open	limited
h. Availability of data for model training and evaluation (e.g., scripts, annotated corpora, metadata)	OK	limited	limited

The suggestion of Table 1 is that the overall complexity of analysing non-scripted content is highly related to the number of interacting speakers, the type of speech (read/non-read) and the availability of training data. The next section will present the current performance of some analysis tools for low-level features that eventually may contribute to a level of understanding for both types of non-scripted that could be called ‘semantic’.

## 4 Performance Evaluation

This section will report on the evaluation for some of the work in M4 and InDiCo. For AMI no performance figures are available as the project started only in 2004.



4.1 M4: Location-Based Speaker Tracking<sup>3</sup>

Automatic annotation of meetings in terms of speaker identities and their locations, which is crucial for the higher level segmentation, is achieved by segmenting the audio recordings using two independent sources of information: magnitude spectrum analysis and sound source localization. We combine the two in an HMM framework. There are three main advantages of this approach. First, it is completely unsupervised, i.e. speaker identities and number of speakers and locations are automatically inferred. Second it is threshold-free, i.e. the decisions are made without the need of a threshold value which generally requires an additional development dataset. The third advantage is that the joint segmentation improves over the speaker segmentation derived using only acoustic features. Experiments on a series of meetings recorded in the IDIAP Smart Meeting Room demonstrate the effectiveness of this approach. For more details, cf. [10].

Table 2. Speaker segmentation performance percentages

Clustering type	HTER	ACC
Acoustic Only	19.2	92.6
Location-based	17.3	94.6

4.2 Speaker Turn Pattern Segmentation

A corpus of meetings recorded and annotated by ICSI [7] was used for the work on speaker turn segmentation by the University of Sheffield. A 60-minute meeting has been segmented using the Bayesian Information Criterion (BIC). This segmentation was compared with manual topic segmentation. Cf. Figure 2.

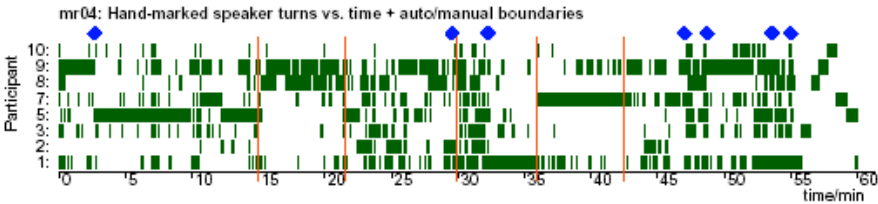


Fig. 2. Each row corresponds to a different participant. The meeting ended with most participants separately reading a series of digits (56 minutes onwards). The 5 vertical lines indicate boundaries from BIC segmentation; 7 (blue) diamonds show the hand-marked topic boundaries for this meeting.

Of 36 manually-marked topic boundaries over 6 meetings, only 15 agreed with turn-based segmentation (margin 2 minutes); in addition, 16 turn-based boundaries were found that had no corresponding topic boundary. It seems turn-pattern boundaries are

<sup>3</sup> Work carried out at IDIAP, Martigny.

not directly related to discussion topics, although they may provide an important alternative perspective on the temporal structure of meetings. For more details, cf. [8].

### 4.3 InDiCo: Speech Recognition<sup>4</sup>

The project has used the SPRACHcore recognition engine, distributed by ICSI, Berkeley.<sup>5</sup> Language models and vocabulary have been optimized with respect to the InDiCo development test database. The Word Error Rate (WER), which was over 80% before optimization, has been reduced to 67%, using a mixed language model based on 400 million words of North American Newspaper texts and the InDiCo document set, 45 million of pre-print texts from the high energy physics domain, from which a vocabulary of 40k words has been derived. The WER figures are likely to improve if a proper pronunciation dictionary, but already are becoming useful for certain retrieval tasks.

### 4.4 InDiCo: Video Analysis<sup>6</sup>

The work on video analysis consists of two tasks: slide segmentation and slide matching. A video containing a slide presentation has to be segmented in such a way that (i) each segment contains no more than one slide, and (ii) a slide that is shown without interruption belongs to one segment only. Via slide matching the slide source can be synchronized with the video time codes. Experiments have been done for two types of slides: (i) *PowerPoint presentations*, which usually have very clear contrast, often colour and clear borders, and (ii) *Printed presentations sheets*, typical for CERN, with vague borders, no colour and illumination varying over the sheet. Results for the former are almost perfect. Results for the latter are still rather poor: on a test set of 8 hours of CERN video data, an accuracy of 65 % was obtained.

### 4.5 InDiCo: Linking Lecture Fragments and Papers<sup>7</sup>

The cross-link utility developed at TNO to support the navigation lecture fragments to slides and from slides to papers is based on machine learning. For every page in a paper, all n-gram word sequences are extracted, which are labelled with the page number. As n-grams implement a shifting window, this produces instances like

```
Deceleration,of,antiprotons,has,been,page_6
of,antiprotons,has,been,demonstrated,page_6
antiprotons,has,been,demonstrated,in,page_6
```

A memory-based learning classifier [9] is strained on these labelled n-grams. For every slide in a presentation, n-grams are extracted along the same lines. The trained memory-based classifier is applied to every n-gram in the slide, and all classifications are gathered. Finally, majority voting is applied to the set of collected predictions, and

<sup>4</sup> Work carried out by TNO Human Factors, Soesterberg.

<sup>5</sup> Cf. <http://www.icsi.berkeley.edu/~dpwe/projects/sprach/>

<sup>6</sup> This work was performed at ISIS, University of Amsterdam.

<sup>7</sup> Work carried out by TNO TPD, Delft.

the class with highest frequency was selected as the winning classification of the slide. The test and training data set consisted of 50 paired papers and slides from the 8th European Particle Accelerator Conference. A link was judged to be correct if (a) it was formally plausible and (b) if it was sequentially plausible, that is: not linked to a page too far from the page the previous slide was linked to. On a representative subset of 8 presentations, approximately 90% accuracy has been achieved.

## 5 Concluding Remarks

The disclosure of recordings of non-scripted events imposes different requirements than fully directed events such as news programs. Due to the low information density of meetings, low-level audio and video features should be exploited for the recognition of high-level meeting actions and event structures that can be indexed and searched for. Metadata for conferences can be enriched by the linking of lecture recordings to collateral linguistic sources. Focus on non-scripted video content may open up new research perspectives for multimedia analysis and novel applications for multimodal information processing, but advances in this domain are highly dependent on proper training and test collections.

## References

- [1] M. Maybury (ed.), "Intelligent Multimedia Information Retrieval", MIT Press, Cambridge (1997)
- [2] Content-Based Access of Image and Video Libraries. Proceedings IEEE Workshop. IEEE Computer Society, Los Alamitos, 1997.
- [3] Human language as media interlingua: Intelligent multimedia indexing. In: Proceedings of ELSNET in Wonderland. ELSNET, Utrecht (1998) 51-57.
- [4] Jong, F. de, Gauvain, J.L., Hartog, J. den, and Netter, K., Olive: Speech-based video retrieval". In Proceedings of CBMI'99. Toulouse (1999) 75-80
- [5] Jong, F. de, Gauvain, J.-L., Hiemstra, D., Netter, K., 2000. Language-Based Multimedia Information Retrieval. In: Proceedings of 6th RIAO Conference. Paris (2000) 713-722
- [6] Smeaton, A.F., W. Kraaij and P. Over, TRECVID -An Introduction, In: Proceedings of TRECVID 2003. Gaithersburg (2003)
- [7] Morgan, N., D. Baron, J. Edwards, et. al., The meeting project at ICSI. In: Proceedings HLT (2001) 246-252
- [8] Renals, S., D. Ellis, Audio Information Access from Meeting Rooms. In: Proceedings IEEE ICASSP 2003 – Hong Kong.
- [9] Daelemans, W., J. Zavrel, K. van der Sloot and A. van den Bosch, TiMBL: Tilburg Memory-Based Learner, version 5.0 (2004). Available at <http://ilk.kub.nl/papers>
- [10] Ajmera, J., G. Lathoud, and I. McCowan, Clustering And Segmenting Speakers And Their Locations In Meetings. In: Proceedings ICASSP (2004)

# A User-Centred System for End-to-End Secure Multimedia Content Delivery: From Content Annotation to Consumer Consumption

Li-Qun Xu<sup>1</sup>, Paulo Villegas<sup>2</sup>, Mónica Díez<sup>2</sup>, Ebroul Izquierdo<sup>3</sup>, Stephan Herrmann<sup>4</sup>, Vincent Bottreau<sup>5</sup>, Ivan Damnjanovic<sup>3</sup>, and Damien Papworth<sup>3</sup>

<sup>1</sup> BT Exact, British Telecommunications PLC, UK

<sup>2</sup> Telefónica I+D, Spain

<sup>3</sup> Queen Mary University of London, UK

<sup>4</sup> Technical University of Munich, Germany

<sup>5</sup> INRIA, France

**Abstract.** The paper discusses the current status of progress of the on-going EU IST BUSMAN project (Bringing User Satisfaction to Media Access Networks), which now approaches the milestone of its 2<sup>nd</sup> year running. The issues explained include the motivation and approaches behind the design of its client-server system architecture for effective data flows handling, the progress in the implementation of the proposed server system functionalities, and the advanced video processing algorithms investigated and adopted. A fully functional client-server system for video content management, search and retrieval for both professional use scenarios and customers with either fixed or wireless network connections is expected to be demonstrable by year end of 2004.

## 1 Introduction

Whilst storage and capture technologies are able to cope with the huge increase in volume and varieties of available video footages, the nature of unstructured data reservoirs makes it an acute problem to access desired pieces of information or contents when and where needed. The structuralisation, or annotation and indexing, of these content data such that it can be effectively managed, searched and retrieved is a must for both businesses and customers' applications, and it is time-consuming and laborious when it is performed manually by human operators. Automated or semi-automatic techniques and tools are urgently sought. Besides, whilst end-users expect an easy access to digital content using terminal devices of varied (networking, computing, storage etc) capabilities and query structures natural and close to human concepts, content creators and providers are looking for efficient and secure distribution systems capable of protecting digital content and tracing illegal copies. The BUSMAN project aims to meet these diverse needs of content creators, providers and end users by designing and implementing an efficient end-to-end system for analysis, annotation, delivery and querying of videos from large databases via heterogeneous networks, whilst complying with the main multimedia standards such as MPEG-7 and MPEG-21.

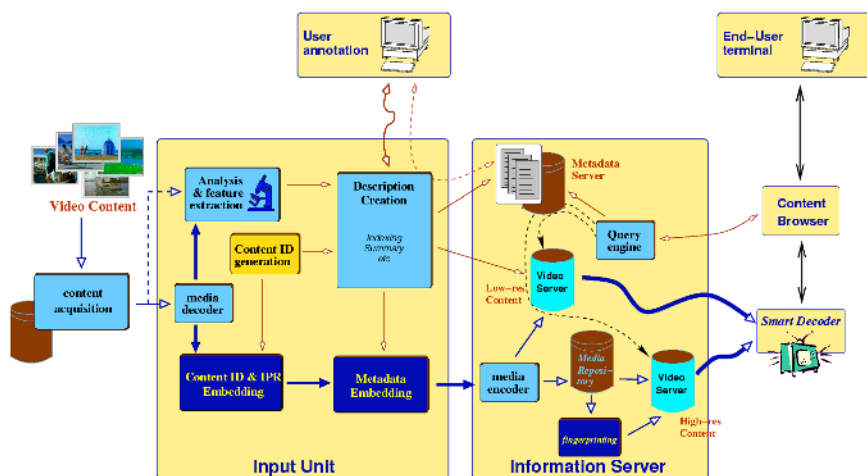


Fig. 1. Busman logical software architecture

Since the launch of the project in April 2002, the consortium has made concerted efforts in addressing some of the key issues behind the development of such a prototype system, including an extensive set of human factors studies (use scenarios, user-centred design, and usability [1]), novel video processing algorithms for content analysis, modularised system design and integrations [2], and user-friendly interfaces. Due to the limited space, this paper only discusses some of the major technical studies and system achievements so far, focussing on logical and functional descriptions.

This paper is organised as follows. In Section 2, the general architecture of the system is presented, which is then followed by discussions on two main aspects of the system, data embedding requirements and techniques (Section 3) and the video analysis, annotation and retrieval modules (Section 4). The paper concludes in Section 5 with a discussion on the remaining works and planned enhancement for the delivery of a fully functional prototype system for video content management.

## 2 System Architecture

We start with illustrating the overall architecture of the BUSMAN conceptual system shown in Figure 1, which follows the popular client-server model.

As it can be seen, the **server** comprises two logical elements: the *Input Unit* (uploading and processing a given media file), and the *Information Server*, which delivers data (content & metadata) to end users. There are two types of **clients**:

- the *annotation client*, which interacts with the input unit, assisting a human annotator who is presumably a professional user; and
- the *end-user terminal*, which connects to the information server for an end user to browse and retrieve desired contents; and for which two versions are being developed including a fixed PC-based desktop terminal and a mobile terminal.

The diagram also shows three different types of data flows along the system, with the filled, empty, and empty diamond arrows describing, respectively, the flow of uncompressed media content, compressed media content, and the metadata extracted or annotated. Figure 2 further describes the processing steps associated with each of the three types of data flows (separated by the two dashed lines) within the Server; the explanation of its functioning mechanism and enabling technologies is the focus of the following discussions.

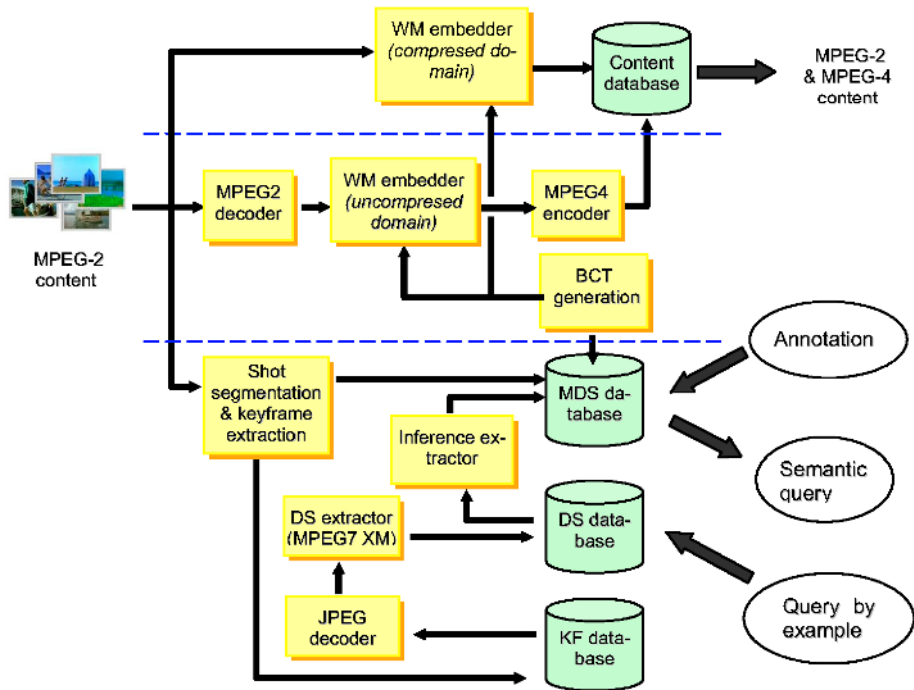


Fig. 2. The three data flows (compressed, uncompressed, metadata) in the BUSMAN server .

### 3 Data Embedding Techniques

The aim of these modules is to provide the system with capabilities of embedding a *Digital Item Identifier* (DII) in the compressed video. A DII can be regarded as a serial number that uniquely identifies a piece of content [3], and therefore can serve conveniently as a pointer to the storage of metadata that describes the syntactic and semantic structures of the video content. For this system, a DII specification called *Busman Content Tag*, or BCT for short, has been defined. Embedding a BCT requires then much less capacity than that needed for full metadata.

Two different data embedding techniques are used in BUSMAN to serve for different purposes, they are outlined as follows:

- The first one is to embed a watermark directly in the MPEG-2 compressed stream, aiming at the professional use scenario defined in the project. The user will be al-

lowed to access metadata by extracting the BCT from the video content, and further on linking the BCT to the database that contains the metadata. Thus, the technical challenges encountered are: embedding and decoding of watermark with high imperceptibility, robustness to video editing processes and fast decoding speed.

To implement it, we use the spread spectrum techniques. Spread spectrum communication transmits a narrow-band signal via a wide-band channel by frequency spreading. In the case of watermarking, it transmits the watermark signal as a narrow-band signal via the video signal acting as the wide-band channel.

In the raw video domain, the watermark signal is usually spread using a large chip factor, modulated by the pseudo-noise sequence, amplified afterwards, and finally added directly to video pixels. Due to performance constraints in terms of computing efficiency and degradation in re-encoding, watermarking in pixel domain is not feasible in our case. For this purpose, after an intermediate parsing of the MPEG-2 stream a watermark signal prepared in the similar fashion is directly added to the DCT coefficients of the compressed video to accomplish the task.

- The second procedure produces MPEG-4 watermarked versions of the video content, destined for streaming to clients. A smart client will be able to extract the BCT, and through it locate the metadata of the content. This watermarking is done in the raw domain, *i.e.* over a sequence of decoded images, taking advantage of the transcoding operation. In order to provide the best possible robustness for the watermark, it is inserted after all video pre-processing tasks (image resizing and/or frame rate adaptation) are completed.

The watermarking component is fed with chunks of images or GoF (*Group of Frames*). The input GoF is spatially and temporally decomposed using 3D (2D+t) wavelet decomposition. The message is then embedded within the low-frequency spatial-temporal coefficients (that represent a kind of spatial-temporal approximation to the input GoF) using a dedicated algorithm. Once the insertion is done, an inverse wavelet transformation is applied to the watermarked spatial-temporal wavelet tree so as to recover the GoF, which is subsequently sent to the MPEG-4 encoder.

By doing this, the message is spread over all the input images of the GoF. Thus, it is reasonable to say that the drop of one frame within a GoF will have little impacts on the watermark detection at the decoder side, provided that the GoF synchronisation is maintained.

## 4 Video Indexing and Retrieval

The metadata processing chain comprises the main modules of video analysis and retrieval subsystem, which takes as input the video content, and produces a range of MPEG-7 compliant metadata, both high-level (semantic) and low-level (media descriptors). The main processing and interfacing modules include shot segmentation and keyframe extraction; shot boundary editing; semantic annotation; semantic search engine and the query-by-example subsystem (low-level video descriptor extraction and descriptor search).

To comply with a standard, the VideoSegmentDS from the MPEG-7 MDS standard [8] was selected to represent the shot information. These descriptions are the central element in the BUSMAN system, because they can store the required meta in-

formation and relate them to temporal units of the content. It can be seen as a synchronisation tool between meta and media information.

## 4.1 Shot Segmentation

Several efficient techniques for shot segmentation have been investigated; two of them have been implemented within the system. They are briefly described in the following.

### 4.1.1 Direct Segmentation from the MPEG-2 Compressed Video Stream

The first option is specially adapted to the MPEG-2 video streams that are used as source material. Thus the *MacroBlock* (MB) type information is conveniently used. The *MBType* defines the character of the MB prediction [4]; there are four prediction types:

- *Intra* coded, MB not predicted at all;
- *Forward* referenced, MB predicted from the previous reference frame;
- *Backward* referenced, MB predicted from the next reference frame;
- *Interpolated*, both reference frames are used to predict the MB in the current frame by interpolation.

Since the MPEG sequence has a high temporal redundancy within a shot, a continuously strong inter-frame reference will be present in the stream as long as no significant changes occur in the scene [5]. The amount of inter-frame references in each frame and its temporal changes can be used to define a metric, which measures the probability of scene change in a given frame. Thus, we can define a metric for the visual frame difference by analysing the statistics of *MBTypes* in each frame.

Since the metrics value is determined separately for each frame and the content change is based on frame triplet element (IBB or PBB), low-pass filtering with kernel proportional to triplet length would eliminate the noise. A filter with Gaussian pulse response is applied to that aim.

### 4.1.2 Segmentation from the Uncompressed Video

The other shot segmentation algorithm integrated into the system derives visual content features from the uncompressed image sequence instead. This places an extra burden on the system (to decode the video sequence), though its impact is not significant except on low-complexity systems. And it has the additional advantage of potentially allowing any video format to be used. The features chosen belong to the well-studied MPEG-7 visual descriptors, thus with known characteristics, such as scalability, small size and low cost matching.

In the current implementation, only one feature vector, the *ColorLayout*, is used. This is a simple DCT-based representation of the layout of colour within a frame. One DCT is created for each colour plane (one luminance and two chrominance components). A distance metric involves taking the weighted Euclidean distance between each DCT value in each colour component. This leads to fast matching; and scalability can be improved by using fewer DCT values at the expense of accuracy.

The resulting feature vector can be used for a variety of applications. Simple shot cuts can be easily detected by locating the peaks in the rate of change of feature be-



tween consecutive frames, which is robust to abrupt shot changes and reasonably accurate even in sequences with high visual activities.

For slower transitions, such as cross fades, the feature differences measured at intervals of 10, 20 and 25 frames apart give characteristic graphs that can be used to detect fades with reasonable levels of accuracy. It is expected that extracting additional features for each frame will help to improve accuracy further.

## 4.2 Shot Boundary Editing

A server application module allows a professional user to modify manually the shot boundaries and keyframes which are automatically created by the system. The interface is shown in Figure 3, and it contains a special timeline with a slider (to move inside the video) and where shot boundaries and keyframes are highlighted. By using the control options, the annotator can change the position of any shot boundary or select as keyframe a different image than the one chosen by the system.

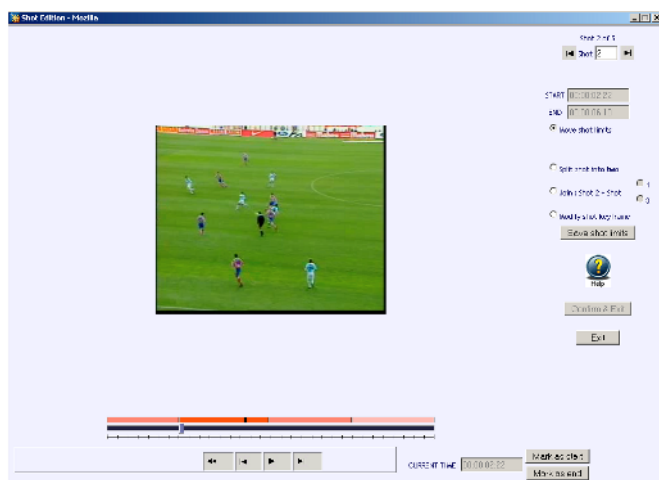


Fig. 3. The interface for shot boundary editing

## 4.3 Semantic Annotation

This application allows a user to attach, edit or delete semantic annotations, which can describe either a complete media file, stating the global semantic descriptions of its content, or a shot, characterising only the semantic content of that shot. The same annotation scheme is adopted for both the two types, though the user interface shows slight differences, of course.

The semantic annotation model used in BUSMAN can be mapped to a subset of the MPEG-7 Multimedia Description Schemes [8,9] standard, chosen as a compromise between expressiveness and ease of annotation. In view of this last requirement

the SemanticDS and friends are discarded as unsuitable to be used by non-expert annotators, since the work needed to create the underlying abstraction model was deemed too complex.

Instead, a simpler and more intuitive keyword-based approach based on the StructuredAnnotation MPEG-7 descriptor [8] is adopted. This descriptor allows storing textual annotations in terms of 7 basic semantic concepts: *animate objects* – *people and animals* (Who), *actions* (WhatObject), *objects* (WhatAction), *places* (Where), *time* (When), *purposes* (Why), and *manner* (How). An annotator can associate any number of these semantic descriptors to a given shot. A dual annotation scheme has been set up by making use of the TermUse MPEG-7 datatype, so that any annotation can be either

- a free text description (but still tagged to one of the aforementioned 7 concepts), or
- a dictionary-based term.

The latter are preferred wherever possible, since they allow more precise annotation, thus providing better query results. Additional effort is made to generalise the use of dictionaries: a dictionary is defined by different qualifiers, allowing associating a term with a set of genres and a given semantic concept (Who, Where, etc).

A hierarchical relationship between dictionaries allows automatic establishment of parent-child relationships between dictionary entries. The whole dictionary structure could be considered as the starting point to further develop a real ontology.

To be able to express more elaborated concepts than just simple keywords, the annotations can be easily clustered within a shot. A cluster of annotations thus defines a set of concepts tied together (e.g., a Who descriptor with a WhatAction descriptor). This is also supported by the MPEG-7 syntax.



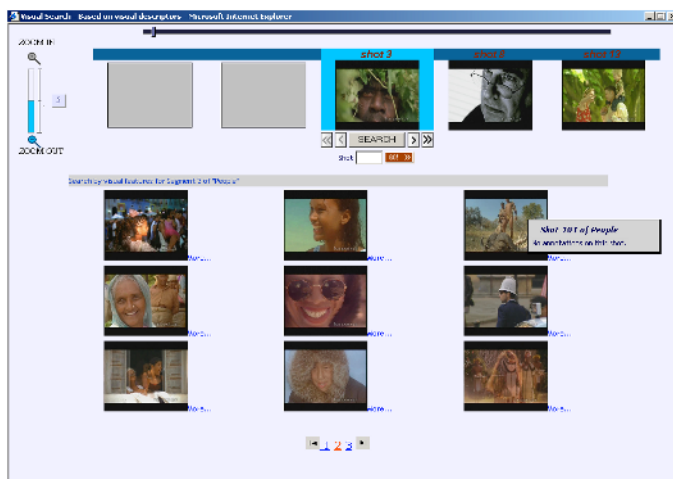
Fig. 4. Interface for shot-based semantic annotation

Figure 4 presents the implemented interface for **shot-based semantic annotations**. A *Shot Browser* with special controls (including a zoom bar) allows locating the desired shot that can then be played in the *Shot Player*. The main section is the *Semantic Annotation* interface that allows easy annotation editing.

Once annotated, users can search for a document and/or a document segment (a shot) with certain semantic annotation(s) among its related metadata. The search interface allows therefore inserting an arbitrary number of query terms (defined as either dictionary entries or free text, both corresponding to the desired annotation type).

#### 4.4 Query by Example

This application module, shown in Figure 5 is its interface, allows a user to search for similar video shots via keyframe matching. For this purpose the user can select query images in the upper part of the window when browsing through the shot keyframes. By clicking on any result image the user can navigate to the shot corresponding to the image, and obtain further metadata information.



**Fig. 5.** The query-by-example search interface

The search is based on measuring a similarity metric of a number of visual MPEG-7 features [7] between the query and candidate images in the database. To extract the visual features and compute the metrics the MPEG-7 XM reference software is used.

Six color and texture MPEG-7 visual descriptors have been used. The selection is based on performance studies. Since the system model does not include any spatial segmentation module, no shape descriptors are used.

At the moment, the distance values of the individual descriptors are combined using a weighted sum. To do this, the distance functions are normalised with respect to their working point (WP). The WP was selected empirically based on experiments, thus, for distance values below the WP, images are perceived to be similar and vice versa. To select the weighting factors in an intuitive way, a relevance feedback function will be added in a future release.

## 5 Discussion

We have presented in this paper a prototype implementation of the BUSMAN system for effective video content management and delivery, laying emphasis on descriptions of the Server side functional modules and the techniques employed. The two main technical aspects explained are the data embedding techniques and the video analysis, annotation, and search subsystem.

The remaining works of the project include investigating tools for the chosen use scenarios and video genres (which may provide automatic suggestions for annotation), and the development of a novel relevance feedback algorithm for query by example search. Also high on agenda are the test and usability studies of the server system and fixed clients. Finally a mobile client [1] is to be fully developed and integrated into the overall system.

**Acknowledgements.** This paper is based on the research and development activities conducted within the IST project BUSMAN which is partially funded by the EU under grant Nr. IST-BUSMAN-2001-35152. URL: [www.ist-busman.org](http://www.ist-busman.org). The contributions from all the partners are acknowledged.

## References

1. A. Evans: User-Centred Design of a Mobile Football Video Database. Proceedings of 2<sup>nd</sup> Intl. Conf. on Mobile and Ubiquitous Multimedia, Norrköping, Sweden, Dec 2003.
2. P. Villegas et al.: An Environment for Efficient Handling of Digital Assets. Proceedings of WIAMIS'2003, London, UK.
3. ISO/IEC JTC1/SC29/WG11, Information Technology – Multimedia Framework – Part 3: Digital Item Identification, ISO/IEC IS 21000-3.
4. D. LeGall, J.L. Mitchell, W.B. Pennbaker, C.E. Fogg: MPEG video compression standard. Chapman & Hall, New York, USA, 1996.
5. J. Calic and E. Izquierdo: Towards Real-Time Shot Detection in the MPEG Compressed Domain. Proceedings of WIAMIS'2001, Tampere, Finland.
6. MPEG Committee Implementation Subgroup: MPEG-7 eXperimentation Model.
7. ISO/IEC JTC1/SC29/WG11, Information Technology – Multimedia Content Description Interface – Part 3: Visual, ISO/IEC IS 15938-3:2001.
8. ISO/IEC JTC1/SC29/WG11, Information Technology – Multimedia Content Description Interface – Part 3: Multimedia Description Schemes, ISO/IEC IS 15938-5.
9. P. Salembier and J. R. Smith: MPEG-7 Multimedia Description Schemes. IEEE Trans. on Circuits and Systems for Video Technology, Vol. 11, No. 6, June, 2001.

# Adding Semantics to Audiovisual Content: The FAETHON Project

Thanos Athanasiadis and Yannis Avrithis

Image, Video and Multimedia Systems Laboratory  
School of Electrical and Computer Engineering  
National Technical University of Athens  
9, Iroon Polytechniou St., 157 73 Zographou, Greece  
{thanos, iavr}@image.ntua.gr

**Abstract.** This paper presents FAETHON, a distributed information system that offers enhanced search and retrieval capabilities to users interacting with digital audiovisual (a/v) archives. Its novelty primarily originates in the unified intelligent access to heterogeneous a/v content. The paper emphasizes the features that provide enhanced search and retrieval capabilities to users, as well as intelligent management of the a/v content by content creators / distributors. It describes the system's main components, the intelligent metadata creation package, the a/v search engine & portal, and the MPEG-7 compliant a/v archive interfaces. Finally, it provides ideas on the positioning of FAETHON in the market of a/v archives and video indexing and retrieval.

## 1 Introduction

In less than ten years the World Wide Web has evolved into a vast information, communication and transaction space. Needless to say its features differ greatly from those of traditional media. Projects and related activities supported under the R&D programs of the European Commission have made significant contributions to developing:

- New models, methods, technologies and systems for creating, processing, managing, networking, accessing and exploiting digital content, including audiovisual (a/v) content.
- New technological and business models for representing information, knowledge and know-how.
- Applications-oriented research – focusing on publishing, audiovisual, culture, education and training – as well as generic research in language and content technologies for all applications areas.

In this framework, the IST project FAETHON [1], has been an approach towards realising the full potential of globally distributed systems that achieve information access and use. Of primary importance is FAETHON's contribution towards the Semantic Web [2]. The fundamental prerequisite of the Semantic Web is “making content machine-understandable”; this happens when content is bound to some formal description of itself, usually referred to as “metadata”. Adding “semantics to content”

in the framework of FAETHON is achieved through algorithmic, intelligent content analysis and learning processes.

FAETHON has closely followed the developments of MPEG-7 [3] and MPEG-21 [4] standardization activities and successfully convolved technologies in the fields of computational intelligence, statistics, database technology, image/video processing, audiovisual descriptions and user interfaces to build, validate and demonstrate a novel intermediate agent between users and audiovisual archives. The overall objective of FAETHON has been to develop a stand-alone, distributed information system that offers enhanced search and retrieval capabilities to users interacting with digital audiovisual archives [5]. The project outcome contributes towards making access to multimedia information, which is met in all aspects of everyday life, more effective and more efficient by providing a user-friendly environment.

This paper is organized as follows: Section 2 presents an overview of the FAETHON system. Section 3 describes the FAETHON intelligent metadata creation package, focusing on the system's offline content processing capabilities. Section 4 deals with the FAETHON a/v search engine and portal, focusing on its advanced semantic search and personalization functionalities. Section 5 describes the FAETHON a/v archive interfaces. Finally, conclusions are drawn in Section 6.

## 2 System Overview

During its conception phase, the overall objective of FAETHON has been decomposed into five primary objectives, namely: (a) extraction of high level semantic information out of existing low-level semantic data, (b) generation and update of user profiles that drive and influence the weighting of the record sets that is queried out of the audiovisual archives, (c) incorporation of the retrieval of high-level semantic information and personalization into generic DBMS schemes, (d) development of intelligent indexing, querying and retrieval mechanisms for effectively handling the huge volumes of data within audiovisual archives and (e) exploitation of the availability of a/v archives, to fine-tune, evaluate, validate and integrate the project's results.

From the end-user point of view, this novel system exploits the advances in handling a/v content and related metadata, as introduced by MPEG-4, and MPEG-7, to offer advanced access services characterized by the tri-fold *semantic phrasing of the request, unified handling* and *personalized response*. From the technical point of view, FAETHON plays the role of an intermediate access server residing between end users and multiple heterogeneous archives, in order to: (a) keep track of user preferences, (b) project these preferences to appropriate indices of the archived content and (c) adjust the responses to user's queries in a manner that fits to his/her priorities.

FAETHON consists of two major components: the *FAETHON Intelligent Metadata Creation Package* that automatically generates content-related metadata to be used for browsing / search / retrieval, and the *FAETHON A/V Search Engine & Portal* that provides end user unified and personalized access to a number of heterogeneous a/v archives. It also contains a number of individual *a/v archive interfaces*, which are responsible for the MPEG-7 compliant communication between the FAETHON search engine and the participating archives, in a way transparent to the end user.

### 3 FAETHON Intelligent Metadata Creation Package

The FAETHON Intelligent Metadata Creation Package manages the FAETHON knowledge and user / archive profiles, analyzing the audiovisual material of a number of heterogeneous audiovisual archives and automatically generating material-related metadata in the form of a semantic index to be used by the FAETHON A/V Search Engine & Portal for unified access to the material. It consists of the following units:

#### 3.1 Encyclopedia Editor

The FAETHON fuzzy relational encyclopedia consists of crisp and fuzzy sets, fuzzy relations and rules concerning the expansion of the fuzzy sets and relations [6,7]. An XML schema has been defined for this purpose, however, the general difficulties that pertain the manipulation of validated XML documents, as well as the need for an automation of the rules and the operation highlighted the necessity of a GUI-based tool.

This tool, the Encyclopedia Editor, has the following functionalities: (a) Graphical manipulation of fuzzy and crisp sets of semantic entities and fuzzy relations. Entities, crisp and fuzzy sets, relations and relation elements can be added, deleted and re-named. (b) Operations on fuzzy sets and fuzzy relations. (c) Fuzzy relation properties are visually listed for each relation. (d) Automatic expansion based on rules stored within the encyclopedia. (e) XML serialization. The encyclopedia editor uses XML to read and store the encyclopedia.

#### 3.2 Description Graph Editor

The Description Graph Editor (DGE) is a graphic user interface that allows the definition of description graphs (DG) for a given semantic entity present in the FAETHON encyclopedia [8]. This definition can rely on previous simpler semantic entities or on MPEG-7 compliant visual descriptors. The system allows defining semantic relations between the semantic entities that form the DG.

The Description Graph Editor has the following additional functionalities: (a) Very simple interaction. (b) Definition of new semantic relations to be used in the creation of future DGs. (c) Definition of a Semantic Entity only based on MPEG7 visual descriptors. (d) Assignment of multiple Description Graphs to the same Semantic Entity, with different level of complexity. (e) Assignment of relevance values to the various nodes (simpler SEs and SRs) defining the Description Graph.

#### 3.3 Detection of Events and Composite Objects in Video Sequences

The Detection of Events and Composite Objects (DECO) unit detects specific composite objects or events in video sequences and related metadata in order to extract their semantic interpretation [1]. It uses the semantic entity definitions (objects, events, concepts) stored in the FAETHON encyclopedia and scans the a/v documents available at the participating archives in order to find and store (as links) the a/v

documents that contain each semantic entity, together with a corresponding degree of confidence. Its output is stored in the FAETHON semantic index to be used by the search engine.

Events and composite objects are Semantic Entities (SEs), which are defined in the FAETHON Encyclopedia by means of the so-called Description Graphs (DGs) and interrelated using Semantic Relations (SRs). Detection and recognition of such entities in video sequences and related metadata is based on DG modeling and representation in order to capture their structure. A generic hybrid neuro-fuzzy architecture has been implemented, which provides rule-based inference and adaptation using the ideas of neural learning and fuzzy inference. The rules for SE recognition are closely coupled to their DG definitions, while state-of-the-art video processing/analysis techniques have been implemented for the detection of fuzzy predicates [9,10,11,12], i.e. extraction of the main mobile objects (and their interrelations) in video sequences, in the general case of sequences acquired by a moving camera. Built-in knowledge about entities and adaptation permits robustness and uncertainty handling, while learning allows updating of knowledge, i.e. adaptation to environmental changes.

### 3.4 Dynamic Thematic Categorization of Multimedia Documents

The Dynamic Thematic Categorization (DTC) unit performs automatic categorization of multimedia documents through matching between the textual metadata associated to the archived material and the thematic categories stored in the FAETHON encyclopedia. It unifies the thematic categories of the a/v archives and scans the a/v documents available at the participating archives in order to find and store (as links) the a/v documents that belong to each thematic category, along with corresponding degrees of confidence [7]. The output of the DTC is stored in the FAETHON semantic index and used for fast retrieval of a/v documents by the search engine.

The FAETHON categorization scheme classifies a document to one or more categories, according to their content [13]. Thus, documents belonging to the same category can be treated similarly, with respect to, e.g., user profiling, document presentation etc. DTC handles thematic categorization, i.e. a categorization in a conceptual level. This categorization uses the semantic entities encountered in a document, in order to classify the latter into classes, such as sports, diplomacy, chemistry and so on. The DTC unit performs a fuzzy hierarchical clustering of the semantic entities [14], relying on knowledge that is stored in the form of semantic relations. The notion of context has a central role in this process. Moreover, the DTC is able to perform categorization based on low-level audiovisual signal processing, using an approach based on decision trees.

### 3.5 User Profile Generation

Personalization of retrieval is the approach that uses the user profiles, additionally to the query, in order to estimate the users' wishes and select the set of relevant documents. In this process, the query describes the user's current search, which is the local interest, while the user profile describes the user's preferences over a long period of time; we refer to the latter as global interest [15]. The User Profile Generation unit generates and updates user preferences based on the usage history. The usage history



is updated after the end of a user query by storing all transactions of the user during the query process. The above transactions characterize the user and express his personal view of the a/v content. The user profile generation unit takes these transactions as input and with the aid of the encyclopedia and the multimedia descriptions of the a/v units referred to in the usage history, extracts the user preferences and stores them in the corresponding user profile.

The user preferences consist of a metadata preferences and a semantic preferences part. The metadata preferences refer to information like creation, media, classification, usage, access, and navigation preferences (e.g. favorite actors / directors or preference for short summaries). The semantic preferences, on the other hand, consist of preferences for semantic entities (interests), as well as preferences for thematic categories [16]. The generation of metadata preferences is implemented using a fuzzy hierarchical clustering and topic extraction technique; the metadata preferences are then used by the Presentation Filtering Unit. Both support unsupervised, fully automatic operation.

## 4 FAETHON A/V Search Engine and Portal

The FAETHON A/V Search Engine & Portal provides to the end user unified and personalized access to a number of heterogeneous audiovisual archives and supporting services including: semantic / metadata search to a/v material, browsing the material based on unified thematic categories, and personalized presentation of a/v material. It consists of the following units:

### 4.1 User Query Analysis Unit

The User Query Analysis Unit performs semantic analysis and interpretation of queries given by the end-user in the form of keyword expressions. Thus, it supports semantic phrasing of the user request in a high, conceptual level. It produces a semantic expression corresponding to each given keyword expression; this semantic expression consists of semantic entities (SEs), and degrees of confidence for each detected SE. In doing so, it uses knowledge stored in the FAETHON encyclopedia. It supports three operations: query interpretation, expansion and personalization.

During query interpretation, the keyword expression is transformed into a corresponding semantic expression, with keywords having been replaced by semantic entities and relations, and a corresponding degree of relevance. During query expansion, the context of the query is automatically detected and then the FAETHON thesaurus is utilized to map each semantic entity found in the semantic expression to a set of entities related to it in the specific context and expand the semantic expression with all the entities related to the initial entities [17,18]. Finally, during query personalization, user preferences are utilized so that the search process is “directed” towards fields in which the documents that satisfy the user request are most likely to be found.

## 4.2 A/V Document Search Engine

The A/V Document Search Engine is responsible for the identification of a/v documents matching the user query by both searching the FEATHON semantic index (semantic search) and communicating with the participating a/v archives (metadata search). For the semantic search, it utilizes the semantic expression produced by the user query analysis unit and produces a list of matching documents, along with the corresponding degrees of confidence [13]. For the metadata search, it constructs and dispatches a unified query to each participating archive, requesting the a/v documents that satisfy the metadata expression contained in the user query. The a/v archive interfaces are responsible for translating and handling this query. The two results produced by the semantic and metadata search from all archives are then combined and the result is a list of a/v document locators matching the user query. This process supports both semantic phrasing of each user request, and unified handling of the request in all archives, in a way transparent to the end user.

## 4.3 A/V Document Classification and Ranking Unit

The A/V Document Classification and Ranking Unit performs ranking (but not filtering) to the a/v documents retrieved by the search engine based on semantic preferences contained within the user profiles. Dynamic categorization and detection of composite entities is performed on the retrieved documents using their entire descriptions and relevance values are assigned after matching with the user preferences [15]. The result is a list of a/v documents with their descriptions, ranked according to the interests that are stored in the user profile. Ranks are updated based on similarity measures, taking into account the degree of confidence of each user preference.

## 4.4 A/V Document Presentation Filtering Unit

The main purpose of the A/V Presentation Filtering Unit is to rank MPEG-7 formatted documents from a/v archives with regard to user preferences that are also stored in MPEG-7 format [7]. Additional functionality allows creating FAETHON specific responses that can be used by the FAETHON System. Through a well-defined and documented interface this component can be used for ranking of documents in other MPEG-7 related systems as well.

## 4.5 End User Interface

The End User Interface consists of the *User Interaction*, *User Presentation* and *User Communication* modules. The FAETHON *User Interaction* module offers a fully operational and functional interface to each user for completing a set of actions through a common web browser. The *User Communication Module* translates queries from the user interaction module according to the definition of the *user query* in the information model and parses the Faethon response for presentation purposes. The user can post a query using the form that consists of the semantic query and metadata fields, such as title, date etc. The *User Presentation Module* dynamically generates

search results from the user queries that are presented in several modes according to the personalization preferences as well as the user access rights. Then, the user has the possibility to see details of the retrieval process, i.e. the semantic entities that matched his textual query (Query Interpretation), the expanded list of semantic entities (Query Expansion), the re-ranked list after the impact of the user’s profile settings and the final ranked list of the documents, as shown in Fig. 1.

SEMANTIC AND METADATA SEARCH - SemanticResponse			
The expanded set of semantic entities has been matched with the following multimedia documents in the Faethon semantic index, with degree of relevance:			
Id	Title	SourceArchive	Score
35	Flugzeugkatastrophe	FAA	0.9
1199	Εταικοπόντες Αιγυόουτου 1974	ERT	0.8
52	Die Vietnamkrise	FAA	0.78
1	Sensationell neuen Rettungsmethode	FAA	0.72
1514	Περικρόμο	ERT	0.68
AVQ-A-004129-0038	Archeological excavations in Rome	Alinari	0.6
11	Ausbau und Elektrifizierung der Strecke Graz-Bruck	FAA	0.5
FCC-F-021960-0000	Exodus of the Belgian population	Alinari	0.45
Pages: << Previous QueryInterpretation QueryExpansion SemanticResponse PresentationResponse ClassificationResponse Next >>			

Fig. 1. Ranked multimedia documents retrieved for a user query with the keyword “politics”. The bar at the bottom indicates the intermediate steps.

5 FAETHON A/V Archive Interfaces

The individual a/v archive interfaces are responsible for the MPEG-7 compliant communication between FAETHON and the participating archive systems, in a way transparent to the end user. Three archive interfaces have been implemented for the three archives participating in FAETHON: FAA (Film Archive Austria), ERT (Hellenic Broadcasting Corporation) and Alinari Archive.

The *FAA archive interface* provides functionality for querying the publicly available data of FAA by SOAP via HTTP. The queries accepted by that interface are MPEG-7 formatted and are translated into the native format of the FAA system. The interface also includes a database access layer, which performs the connection to FAA’s ORACLE database.

The ERT (Hellenic Broadcasting Corporation) archive utilizes the MPEG-7 content description standard and consists of two servers, namely (i) the database server, which hosts the archive database with all a/v content metadata, and (ii) the media server / web server / web service provider, which hosts all the a/v content itself, handles media streaming, provides an end-user web interface to the archive content, and serves as a web service provider to interface with the central FAETHON system. For the communication between ERT archive and FAETHON central system structured XML is used, which the *ERT archive interface* produces by parsing and translating the user query. Then an additional database access layer performs the search in ERT database, produces the response, assembles it into an MPEG-7 compliant format and returns it to FAETHON central system. This process makes access to the ERT archive transparent to the end user.

Finally, the *Alinari archive interface* establishes the communication between the FAETHON system and the Alinari Archive system. Additionally, Alinari interface

poses the user query to the database and returns the results. Same as above, all communication between system's modules is achieved with MPEG-7 compliant format in XML, while the whole procedure of the communication of FAETHON with Alinari archive remains transparent to the user.

## 6 Conclusions

The key aspect of the FAETHON developments has been the generation and use of metadata in order to provide advanced content management and retrieval services. The Web will change drastically in the following years and become more and more multimedia enabled, making already complex content management tasks even more complex and requiring solutions based on Semantic Web technologies. Unlike today, content itself will be a commodity in a future Web, making the use of metadata essential. Content providers for instance will have to understand the benefits obtained from the systematic generation of metadata; service providers will have to accept metadata as the basis on which to build new services; and the producers of software tools for end-users will redirect their imagination towards more appropriate integration of application software with Web content, taking advantage of metadata. These developments clearly present some challenging prospects, in technological, economic, standardisation and business terms.

Another interesting perspective of FAETHON's developments is the personalisation, based on usage history, of the results of content retrieval. Personalisation software is still in its infancy, which means there are no turnkey solutions. Solutions using agent technologies still have a lot of hurdles to overcome. To improve this scenario, additional technology approaches need to be evaluated and areas of improvement identified. In both perspectives, clearly FAETHON made some interesting steps on the correct route and its developments are currently influencing the next research activities in the area of semantic based knowledge systems [19].

## References

1. G. Stamou, Y. Avrithis, S. Kollias, F. Marques and P. Salembier, "Semantic Unification of Heterogenous Multimedia Archives", Proc. of 4th European Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2003), London, UK, April 9-11, 2003.
2. J. Hunter, "Adding Multimedia to the Semantic Web: Building an MPEG-7 Ontology", in Proc. of First Semantic Web Working Symposium, SWWS'01, Stanford University, California, USA, July 2001.
3. T. Sikora, "The MPEG-7 Visual standard for content description - an overview", IEEE Trans. on Circuits and Systems for Video Technology, special issue on MPEG-7, 11(6):696-702, June 2001.
4. ISO/IEC JTC1/SC29/WG11 N5231, "MPEG-21 Overview", Shanghai, October 2002.
5. Akrivas, G, Stamou, G, "Fuzzy semantic association of audiovisual document descriptions", Proc. of Int. Workshop on Very Low Bitrate Video Coding (VLBV01), Athens, Greece, October 2001.

6. Akrivas, G., Stamou, Stefanos Kollias, "Semantic Association of Multimedia Document Descriptions through Fuzzy Relational Algebra and Fuzzy Reasoning", *IEEE Transactions on Systems, Man, and Cybernetics*, part A, Volume 34 (2), March 2004.
7. Y. Avrithis, G. Stamou, M. Wallace, F. Marques, P. Salembier, X. Giro, W. Haas, H. Val-lant, and M. Zufferey, "Unified Access to Heterogeneous Audiovisual Archives", *Proc. of 3rd International Conference on Knowledge Management (IKNOW '03)*, Graz, Austria, July 2-4, 2003.
8. X. Giró, F. Marqués, "Semantic entity detection using description graphs", *Workshop on Image Analysis for Multimedia Services WIAMIS-03*, pp. 39-42, London, April 2003.
9. G. Tsechpenakis, Y. Xirouhakis and A. Delopoulos, "A Multiresolution Approach for Main Mobile Object Localization in Video Sequences", *International Workshop on Very Low Bitrate Video Coding (VLBV01)*, Athens, Greece, October 2001.
10. Tsechpenakis, N. Tsapatsoulis and S. Kollias, "Probabilistic Boundary-Based Contour Tracking with Snakes in Natural Cluttered Video Sequences", *International Journal of Image and Graphics: Special Issue on Deformable Models for Image Analysis and Pattern Recognition*, to appear.
11. J. Ruiz and P. Salembier, Robust segmentation and representation of foreground key-regions in video sequences, *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'01)*, Vol. III: IMDSP-L5, Salt Lake City, Utah, USA, May 7-11, 2001.
12. V. Vilaplana, F. Marqués, "Join detection and segmentation of human faces in color images", *Proceedings of the EUROIMAGE ICAV3D 2001, International Conference on Augmented, Virtual Environments and three dimensional imaging*, pp.347-350, Mikonos, Grecia, Mayo del 2001.
13. Wallace, M., Akrivas, G., Mylonas, P., Avrithis, Y., Kollias, S. "Using context and fuzzy relations to interpret multimedia content", *Proceedings of the Third International Workshop on Content-Based Multimedia Indexing (CBMI)*, IRISA, Rennes, France, September 2003.
14. Wallace, M., Akrivas, G. and Stamou, G., "Automatic Thematic Categorization of Documents Using a Fuzzy Taxonomy and Fuzzy Hierarchical Clustering", *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, St. Louis, MO, USA, May 2003.
15. Wallace, M., Akrivas, G., Stamou, G. and Kollias, S., "Representation of user preferences and adaptation to context in multimedia content-based retrieval", *Proceedings of the Workshop on Multimedia Semantics, SOFSEM 2002: Theory and Practice of Informatics*, Milovy, Czech Republic, November 2002.
16. Wallace, M. and Stamou, G., "Towards a Context Aware Mining of User Interests for Consumption of Multimedia Documents", *Proceedings of the IEEE International Conference on Multimedia (ICME)*, Lausanne, Switzerland, August 2002.
17. Akrivas, G., Wallace, M., Andreou, G., Stamou, G. and Kollias, S., "Context-Sensitive Semantic Query Expansion", *Proceedings of the IEEE International Conference on Artificial Intelligence Systems (ICAIS)*, Divnomorskoe, Russia, September 2002.
18. Akrivas, G., Wallace, M., Stamou, G. and Kollias, S., "Context-Sensitive Query Expansion Based on Fuzzy Clustering of Index Terms", *Proceedings of the Fifth International Conference on Flexible Query Answering Systems (FQAS)*, Copenhagen, Denmark, October 2002.
19. I. Kompatsiaris, Y. Avrithis, P. Hobson and M.G. Strinzis, "Integrating Knowledge, Semantics and Content for User-Centred Intelligent Media Services: the aceMedia Project", in *Proc. of Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS '04)*, Lisboa, Portugal, April 21-23, 2004.

# Towards a Large Scale Concept Ontology for Broadcast Video

Alexander G. Hauptmann

School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA  
alex+@cs.cmu.edu

**Abstract.** Earlier this year, a major effort was initiated to study the theoretical and empirical aspects of the automatic detection of semantic concepts in broadcast video, complementing ongoing research in video analysis, the TRECVID video analysis evaluations by the National Institute of Standards (NIST) in the U.S., and MPEG-7 standardization. The video analysis community has long struggled to bridge the gap from successful, low-level feature analysis (color histograms, texture, shape) to semantic content description of video. One approach is to utilize a set of intermediate textual descriptors that can be reliably applied to visual scenes (e.g. outdoors, faces, animals). If we can define a rich enough set of such intermediate descriptors in the form of large lexicons and taxonomic classification schemes, then robust and general-purpose semantic content annotation and retrieval will be enabled through these descriptors.

Our efforts are substantially broad, as our subject matter is broadcast video, which is almost much unrestricted in terms of content, but includes audio and spoken dialog. In addition, broadcast video has an added layer of 'editing': shots and scenes which are carefully chosen to make the point in a broadcast, but do not directly reflect a reality like surveillance video. We are exploring to what extent broadcast video is amenable to a structured characterization of content (using a large but well-defined lexicon). By necessity, the lexicon will have to be general and broadly applicable, since it will be impossible to give in-depth characterizations of shots for the broad content matter that we are dealing with.

But what are the appropriate lexicon items that would allow a sufficiently rich and general description of the video content in broadcast news, which in effect would constitute a general-purpose ontology for describing video content? Our first challenge is to find a large, broad set of descriptors which will be useful in a large variety of broadcast news content. Together with librarians, video archive specialists and experts in multimedia analysis and knowledge representation, we are attempting the definition of an ontology and creating a vocabulary of about a thousand lexical terms that describe the content of broadcast video. Once we have such semantic descriptors, the next question is whether they are actually useful. We will explore the sufficiency and generality of this set of descriptors for annotation and retrieval of video content. One aspect of this work is to empirically determine the feasibility of automatically identifying these descriptions in appropriate video content. We will also annotate

larger amounts of video to see if the set of derived descriptors is appropriate over a wide range of content, and to provide a reference truth for an annotated video library.

Finding 1000 concepts represented in broadcast news video that can be detected and evaluated necessitates careful lexicon design. The concepts in the lexicon should be useful from a perspective of visual information exploitation. Simultaneously the lexicon must be feasible from the perspective of automatic and semi-automatic detection. The design of the lexicon thus needs to bring together members of the library sciences community, knowledge representation as well as researchers from the multimedia analysis community. The confluence of statistical and non-statistical media analysis with ontologies, classification schemas and lexicons helps place the scalable multimedia semantic concept detection problem in the proper context. Context-sensitive concept detection can also help enhance the detection performance and help the scalability. In designing and evaluating a large scale lexicon the following challenges need to be tackled:

- Interpretation of user needs, finding out what do users want from video archives of broadcast news.
- Rigorous experiments to understand how user needs can be mapped into the components of the lexicon
- Study of automatic concept detection system performance and their impact on retrieval performance and classification of concepts on the basis of performance and relevance.
- Understanding of algorithmic approaches for large scale concept detection.
- Empirical and theoretical study of the impact of detection performance trade-off on the ultimate usability of the lexicon, specifically evaluating detection accuracy vs. retrieval performance.

A year-long workshop is under way, developing recommendations and general criteria for the design of large-scale lexicons for audio-visual content classification in support of systems for searching, filtering, and mining of broadcast video. Our approach is to start with a large, fixed collection of data and explore different types of annotation and lexical labeling for retrieval and description. The resulting lexicon and ontology, if successful, will provide a basis for generations of broadcast news video retrieval and annotation work.

# Author Index

- Addis, Matthew J. 638  
Athanasiadis, Thanos 555, 665  
Avrithis, Yannis 555, 665
- Bae, Tae Meon 401  
Baeza-Yates, R. 189  
Bai, Liang 98  
Baillie, Mark 70  
Barroso, B. 500  
Belkin, Nicholas J. 5  
Berretti, S. 464  
Bertolotto, Michela 535  
Bimbo, A. Del 464  
Boldareva, Liudmila 308  
Boniface, Mike J. 638  
Bose, Prosenjit 410  
Bottreau, Vincent 656  
Bouthemy, Patrick 419  
Briggs, Pam 628  
Bruno, Eric 384  
Burford, Bryan 628
- Cai, Rui 79  
Ćalić, Janko 601  
Campbell, Neill 601  
Canagarajah, Nishan 601  
Cao, Yu 160  
Carrara, Paola 517  
Castillo, C. 189  
Chang, Shih-Fu 1  
Chen, Jianyun 98  
Chen, Ming-yu 270  
Christodoulakis, Stavros 582  
Chua, Tat-Seng 545  
Chun, Seong Soo 261  
Chung, Min Gyo 448  
Clough, Paul 243  
Contreras, Jesus 610
- Dahyot, R. 88  
Damjanovic, Ivan 656  
Declerck, Thierry 610  
Detyniecki, Marcin 473  
Diakopoulos, Nicholas 299  
Díez, Mónica 656  
Dorado, Andres 199
- Doulaverakis, Haralambos 592  
Duygulu, Pinar 132
- Eakins, John P. 141, 628  
Essa, Irfan 299  
Everingham, Mark 289  
Ewerth, Ralph 216
- Fan, Jianping 365, 374  
Fauvet, Brigitte 419  
Feng, Huamin 545  
Fitzgibbon, Andrew 2  
Fonseca, Manuel J. 500  
Freisleben, Bernd 216  
French, James C. 252
- Gao, Yuli 365, 374  
García, Ana 225  
Garg, Ashutosh 353  
Gatica-Perez, Daniel 150  
Giorgini, Fabrizio 638  
Gllavata, Julinda 216  
Goodall, Simon 638  
Groen, Piet C. de 160  
Gros, Patrick 419
- Han, Zhi-Guang 115  
Hare, Jonathon S. 317  
Hauptmann, Alexander G. 60, 132, 270, 674  
Heesch, Daniel 491  
Henrich, Andreas 455  
Herranz, Luis 225  
Herrmann, Stephan 592, 656  
Hiemstra, Djoerd 308  
Hirota, Kaoru 51  
Hohl, Lukas 564  
Hollink, L. 6  
Houten, Ynze van 15  
Howarth, Peter 326  
Huang, Thomas S. 353  
Huet, Benoît 483, 564  
Hurtado, C. 189  
Hussain, Mustaq 141



- Ide, Ichiro 123  
 Izquierdo, Ebroul 199, 656  
  
 Jain, Ramesh 299  
 Jeon, Jiwoon 24  
 Jin, Sung Ho 401  
 Jin, Xiangyu 252  
 Jing, Feng 438  
 Jong, Franciska de 647  
 Jorge, Joaquim A. 500  
 Jose, Joemon M. 70  
  
 Katayama, Norio 123  
 Kim, Hyeokman 261  
 Kim, Jung-Rim 261  
 Kim, Minhwan 393  
 Kim, Sungyoung 393  
 Kim, Whoi-Yul 170  
 Koelma, D.C. 6  
 Kokaram, A. 88  
 Kompatsiaris, Ioannis 592  
 Koskela, Markus 234, 508  
 Kuper, Jan 610  
  
 Laaksonen, Jorma 234, 508  
 Laborde, Ron 601  
 Laganriere, Robert 410  
 Lahanier, Christian 638  
 Lao, Song-Yang 98, 106, 115  
 Lee, Chin-Hui 545  
 Lee, Jae-Ho 170  
 Lewis, Paul H. 317, 638  
 Li, Dalei 160  
 Li, Mingjing 438  
 Li, Yunhao 98  
 Loui, Alexander 150  
 Luan, Xi-Dao 106, 115  
 Luo, Hangzai 365, 374  
  
 Manmatha, R. 24, 42  
 Marchand-Maillet, Stéphane 384  
 Marks, Joe 3  
 Martin, W.N. 252  
 Martínez, José M. 225  
 Martinez, Kirk 638  
 Matinmikko, Esa 234  
 McLoughlin, Eoin 535  
 Medina Beltran de Otalora, Raul 592  
 Merialdo, Bernard 483, 564  
 Metzler, Donald 42  
  
 Mezaris, Vasileios 573, 592  
 Mirmehdi, Majid 601  
 Missaoui, Rokia 335, 428  
 Miyamori, Hisashi 179  
 Mo, Hiroshi 123  
 Moënné-Loccoz, Nicolas 384  
 Müller, Henning 243  
 Müller, Wolfgang 455  
  
 Nguyen, G.P. 6  
 Nguyen, Hieu T. 33  
  
 O'Sullivan, Dymrna 535  
 Odobez, Jean-Marc 150  
 Oh, JungHwan 160  
 Oh, Sangwook 448  
 Oja, Erkki 234, 508  
 Ojala, Timo 234  
 Omhover, Jean-Francois 473  
  
 Pala, P. 464  
 Palenichka, Roman M. 428  
 Papageorgiou, Harris 619  
 Papworth, Damien 656  
 Park, Sojung 393  
 Pasi, Gabriella 517  
 Pepe, Monica 517  
 Polydoros, Panagiotis 582  
 Porter, Sarah 601  
 Protopapas, Athanassios 619  
  
 Rampini, Anna 517  
 Rasheed, Zeeshan 279  
 Rautiainen, Mika 234  
 Rea, N. 88  
 Ribeiro, P. 500  
 Rijsbergen, Cornelis J. van 70  
 Ro, Yong Man 401  
 Rüger, Stefan 326, 491  
 Ruiz-del-Solar, J. 189  
  
 Saggion, Horacio 610  
 Samiotou, Anna 610  
 Sanderson, Mark 243  
 Sarifuddin, M. 335  
 Satoh, Shin'ichi 123  
 Schiele, Bernt 207  
 Schreiber, A.T. 6  
 Schuurman, Jan Gerrit 15  
 Seigneur, Jean-Marc 526  
 Shah, Mubarak 279

Shevlin, Fergal 526  
 Shi, Rui 545  
 Sinclair, Patrick A.S. 638  
 Smeulders, Arnold 33  
 Solis, Daniel 526  
 Souvannavong, Fabrice 483, 564  
 Spindler, Fabien 419  
 Stefi, Teuta 216  
 Stejić, Zoran 51  
 Stevenson, James 638  
 Strintzis, Michael G. 573, 592  
 Sull, Sanghoon 261, 448  
 Sun, Ming-Ting 150  
  
 Takama, Yasufumi 51  
 Tavanapong, Wallapak 160  
 Thomas, Barry T. 601  
 Triroj, Napat 150  
 Tsinaraki, Chrisa 582  
  
 Vaillancourt, Jean 335  
 Verhagen, Pløn 15  
 Verschae, R. 189  
 Villegas, Paulo 656  
 Vogel, Julia 207  
 Vries, Arjen P. de 344

Wallace, Manolis 555  
 Wang, Peng 79  
 Wen, Jun 106  
 Westerveld, Thijs 344  
 Whitehead, Anthony 410  
 Wilson, David 535  
 Wittenburg, Peter 610  
 Wong, Johnny 160  
 Worring, M. 6  
 Wu, Ling-Da 98, 106, 115  
  
 Xiao, Peng 106, 115  
 Xie, Yu-Xiang 106, 115  
 Xu, Guangyou 365, 374  
 Xu, Li-Qun 656  
  
 Yan, Rong 60  
 Yang, Jun 270  
 Yang, Shi-Qiang 79  
 Yoon, Ja-Cheon 261  
  
 Zaremba, Marek B. 428  
 Zhai, Yun 279  
 Zhang, Bo 438  
 Zhang, Hong-Jiang 438  
 Zhou, Xiang Sean 353  
 Zisserman, Andrew 289